REPORT DOCUMENTATION PAGE	*Form Approved* **OMB No. 0704-0188**

1. REPORT DATE *(DD-MM-YYYY)* 10-FEB-2010	**2. REPORT TYPE** Final Technical Report	**3. DATES COVERED** *(From-to)* 04-MAR-02 to 05-MAY-08

4. TITLE AND SUBTITLE Higher-Dimensional Signal Processing via Multiscale Geometric Analysis; [original title before award renewal]:Coherent Multiscale Statistical Modeling using Complex Wavelets	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER N00014-02-1-0353
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Baraniuk, Richard G., and Choi, Hyeokho	**5d. PROJECT NUMBER**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Rice University 6100 Main St. MS 16 Houston TX 77005	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Reza Malek-Madani Office of Naval Research, Code 311 875 N Randolph St. Arlington, VA 22203	**10. SPONSOR/MONITOR'S ACRONYM(S)** ONR
	11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT This project pursued a general theory for complex-valued multiscale signal and image modeling, processing, and analysis that is matched to singularity-rich data. Higher-dimensional signals that feature geometric manifold structures were of particular interest in developing theory and a practical toolset for analysis and processing. We pursued a three-pronged approach in creating new multiscale transforms, new geometric statistical models, and new manifold-based signal representations. The results of our research include (1) the Dual Tree Quaternion Wavelet, an efficient transform and analysis tool that features near shift invariance and linear computational complexity; (2) a geometric hidden Markov tree wavelet model, which accounts for geometric regularity by capturing the dependencies between complex wavelet coefficients along a contour; and (3) surflet representations of signal discontinuities with near-optimal rate-distortion performance. These new tools have led to significant performance gains immediately applicable to a number of important Navy-relevant applications, including target detection and classification, image segmentation and fusion, and computer network traffic modeling.

15. SUBJECT TERMS
complex wavelets, multiresolution, coherent processing, edges, graphical models, surflets, feature extraction, geometry, dual-tree, multiscale disparity estimation, phase, quaternion, wavelets

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT** SAR	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Richard G. Baraniuk
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U			**19b. TELEPONE NUMBER (*Include area code)*** 713-348-5132

Higher-Dimensional Signal Processing via Multiscale Geometric Analysis Final Report

Richard G. Baraniuk and Hyeokho Choi

This project pursued a general theory for complex-valued multiscale signal and image modeling, processing, and analysis that is matched to singularity-rich data. Higher-dimensional signals that feature geometric manifold structures were of particular interest in developing theory and a practical toolset for analysis and processing. We pursued a three-pronged approach in creating new multiscale transforms, new geometric statistical models, and new manifold-based signal representations. The results of our research include (1) the Dual Tree Quaternion Wavelet, an efficient transform and analysis tool that features near shift invariance and linear computational complexity; (2) a geometric hidden Markov tree wavelet model, which accounts for geometric regularity by capturing the dependencies between complex wavelet coefficients along a contour; and (3) surflet representations of signal discontinuities with near-optimal rate-distortion performance. These new tools have led to significant performance gains immediately applicable to a number of important Navy-relevant applications, including target detection and classification, image segmentation and fusion, and computer network traffic modeling.

1 Introduction

This is the final report for ONR Grant N00014-02-1-0353 *Coherent Multiscale Statistical Modeling using Complex Wavelets* and its one-year competitive renewal, *Higher-Dimensional Signal Processing via Multiscale Geometric Analysis*. We begin by restating the motivation for our work, reviewing the project objectives, and summarizing our work. We present our results and follow each research thrust with potential future areas of work. We conclude with a list of publications supported by the grant, and a list of project personnel.

1.1 Review of motivation

Over the past twenty years multiscale methods like the discrete wavelet transform (DWT) have revolutionized signal processing. Wavelets are particularly apt for representing singularity-rich one-dimensional (1-D) signals. In contrast to the sinusoidal basis functions underlying Fourier analysis, wavelet basis functions, or atoms, are localized in both space and frequency and thus yield sparse and structured representations of piecewise smooth signals (signals that are smooth except for a finite number of localized point singularities or jump discontinuities). This sparsity and structure boost the performance of wavelet-domain statistical models and enable simple yet powerful algorithms for estimation/denoising, compression, classification, and segmentation.

Despite the remarkable advantages of wavelets for analyzing and processing l-D signals, a surprising realization of the past few years is the inability of wavelets to capitalize in a similar way on 2-D images containing sharp edges. The new, confounding aspect of edge singularities is geometry: the jump discontinuity making up a typical edge is localized along a smooth l-D contour in 2-D. The DWT is incapable of exploiting the fact that the contour is smooth and thus predictable, and as a result it takes too many wavelets to build up the edge, destroying any hope of sparsity and optimality for image processing.

The response from the computational harmonic analysis (CHA) community has been a new generation of multiscale geometric analysis (MCIA) toolsfor 2-D piecewise smooth functions, including wedgelets, ridgelets, curvelets, contourlets, bandelets, wedgeprints, and complex wavelets. In order to efficiently deal with the local geometrical structure of image edges, these representations feature atoms that are both local and directional (oriented). This programme in 2-D has been quite successful; most of the above tools provide optimal representations for piecewise smooth images with smooth ($\mathcal{C}^2$) edges.

However, progress has not continued apace in higher dimensions (3-D, 4-D, and beyond) even though there are an increasing number of important and emerging applications in 3-D (video; computer graphics and modeling; volumetric, hyperspectral, seismic, and astronomical imaging and remote sensing; and robotics), 4-D (light-field imaging, time-lapse seismic imaging), and 5-D (plenoptic modeling, light-field video, virtual reality). The reason for this lack of progress is that the geometrical structure of singularities in 3-D and above is not merely a simple extension of that of edge singularities in 2-D. The added complexity in higher- dimensions is that singularities can themselves be multidimensional. Consider as a prototype n-D piecewise smooth signals that are smooth everywhere except for singularities along smooth m-D manifolds (rods, sheets, and so on), with $0 \leq m \leq n$. For instance, an edge discontinuity in a 2-D image lies along a l-D curve; an aircraft flying through 3-D space tracks out a smooth 1-D curve in 4-D space-time; a video of a smooth object moving smoothly in time lies on a 2-D manifold in 3-D space-time; and points in space can be viewed as lying on 0-D manifolds in 3-D space.

Manifold singularity structures are local and have a preferred orientation. they are smooth in the "direction" of the manifold (along the m-D tangent space) and contain a singularity (rapid change) in the "direction" of the normal (along the $(n-m)$-D normal space). In order to carry the success of MGA into higher dimensions, there is a great need for new theory and tools to exploit these geometric structures. Other than a few scattered promising results, such theory and tools do not exist today.

1.2 Project overview

1.2.1 Project objectives and accomplishments

This project pursued a general theory for complex-valued multiscale signal and image modeling, processing, and analysis that is matched to singularity-rich data. Higher-dimensional signals that feature geometric manifold structures were of particular interest in developing theory and a practical toolset for analysis and processing. The three primary thrusts of our research were:

- **New multiscale transforms:** We attacked a specific and important drawback of current wavelet grammars, the fact that virtually all current models deal exclusively with the energy or *magnitude* of the wavelet coefficients. To more accurately, realistically, and efficiently represent signal singularity structures, we proposed the creation of new complex wavelet transforms and analysis tools. The result was our Dual Tree Quaternion Wavelet, an efficient transform and analysis tool that features near shift invariance and linear computational complexity.

- **New geometric statistical models:** Traditional wavelet-based image processing algorithms and models have significant shortcomings in their treatment of edge contours. Our goal in creating new geometric statistical models was to leverage both magnitude and phase behavior of the complex wavelet vocabulary to more better represent signal singularities. Our work on this project thrust produced a geometric hidden Markov tree wavelet model. The model accounts for geometric regularity by capturing the dependencies between complex wavelet coefficients along a contour, and immediately led to a feature extraction application.

- **New manifold-based signal representations:** We wanted to explore representations of higher dimensional signals whose locality and directionality were matched to the geometry of lower-dimensional manifold structures. In so doing our aim was to better represent signal discontinuities, an important aspect of many signal processing applications. Our work focused on introducing a surflet representation for N-dimensional Horizon functions containing a $\mathcal{C}^K$ smooth singularity in $N-1$ dimensions. Surflets allowed a multiscale, piecewise polynomial approximation of discontinuities. We also created a compression algorithm using surflets that comes within a logarithmic factor of the optimal rate-distortion performance for this function class.

1.2.2 Research impact

Our new models and analysis open unprecedented possibilities for designing processing tools matched to singularity-rich data. These new tools have led to significant performance gains immediately applicable to a number of important Navy-relevant applications, including target detection and classification, image segmentation and fusion, and computer network traffic modeling. Our approach generalizes to other applications as well, including machine fault monitoring, medical imagery, robotics, virtual reality, remote sensing, and sensor networks.

2 Dual Tree Quaternion Wavelets: A new multiscale transform

2.1 Summary of results

The dual-tree quaternion wavelet transform (QWT) is a new multiscale analysis tool for geometric image features. The QWT is a near shift-invariant tight frame representation whose coefficients sport a magnitude and three phases:

two phases encode local image shifts while the third contains image texture information. The QWT is based on an alternative theory for the 2-D Hilbert transform and can be computed using a dual-tree filter bank with linear computational complexity. To demonstrate the properties of the QWT's coherent magnitude/phase representation, we develop an efficient and accurate procedure for estimating the local geometrical structure of an image. We also develop a new multiscale algorithm for estimating the disparity between a pair of images that is promising for image registration and flow estimation applications. The algorithm features multiscale phase unwrapping, linear complexity, and sub-pixel estimation accuracy.

2.2 Review of Real and Complex Wavelets

2.2.1 Real DWT

The real DWT represents a 1-D real-valued signal $f(t)$ in terms of shifted versions of a *scaling function* $\phi(t)$ and shifted and scaled versions of a *wavelet function* $\psi(t)$ [1]. The functions $\phi_{L,p}(t) = 2^L\phi(2^L t - p)$ and $\psi_{\ell,p}(t) = 2^\ell\psi(2^\ell t - p)$, $\ell \geq L, p \in \mathbb{Z}$ form an orthonormal basis, and we can represent any $f(t) \in L_2(\mathbb{R})$ as

$$f(t) \quad = \quad \sum_{p\in\mathbb{Z}} c_{L,p}\phi_{L,p}(t) + \sum_{\ell\geq L, p\in\mathbb{Z}} d_{\ell,p}\psi_{\ell,p}(t), \tag{1}$$

where $c_{L,p} = \int f(t)\phi_{L,p}(t)dt$ and $d_{\ell,p} = \int f(t)\psi_{\ell,p}(t)dt$ are the scaling and wavelet coefficients, respectively. The parameter L sets the coarsest scale space that is spanned by $\phi_{L,p}(t)$. Behind each wavelet transform is a filterbank based on lowpass and highpass filters.

The standard real 2-D DWT is obtained using tensor products of 1-D DWTs over the horizontal and vertical dimensions. The result is the scaling function $\phi(x)\phi(y)$ and three subband wavelets $\phi(x)\psi(y)$, $\psi(x)\phi(y)$, and $\psi(x)\psi(y)$ that are oriented in the horizontal, vertical, and diagonal directions, respectively [1].

The real wavelet transform suffers from *shift variance*; that is, a small shift in the signal can greatly perturb the magnitude of wavelet coefficients around singularities. It also lacks a notion of phase to encode signal location information and suffers from aliasing [2]. These issues complicate modeling and information extraction in the wavelet domain.

2.2.2 Dual-tree CWT

The 1-D dual-tree CWT expands a real-valued signal in terms of *two* sets of wavelet and scaling functions obtained from two independent filterbanks [3], as shown in Figure 1. We will use the notation $\phi_h(t)$ and $\psi_h(t)$ to denote the scaling and wavelet functions and $c_{h_{L,p}}$ and $d_{h_{\ell,p}}$ to denote their corresponding coefficients, where h specifies a particular set of wavelet filters. The wavelet functions $\psi_h(t)$ and $\psi_g(t)$ from the two trees play the rôle of the real and imaginary parts of a complex analytic wavelet $\psi^c(t) = \psi_h(t) + j\psi_g(t)$. The imaginary wavelet $\psi_g(t)$ is the 1-D HT of the real wavelet $\psi_h(t)$. The combined system is a 2× redundant tight frame that, by virtue of the fact that $|\psi^c(t)|$ is non-oscillating, is near shift-invariant.[1]

It is useful to recall that the Fourier transform of the imaginary wavelet $\Psi_g(\omega)$ equals $-j\Psi_h(\omega)$ when $\omega > 0$ and $j\Psi_h(\omega)$ when $\omega < 0$. Thus, the Fourier transform of the complex wavelet function $\Psi_h(\omega) + j\Psi_g(\omega) = \Psi^c(\omega)$ has no energy (or little in practice) in the negative frequency region,[2] making it an analytic signal [3].

2.2.3 Hilbert transforms and 2-D CWT

Extending the 1-D CWT to 2-D requires an extension of the HT and analytic signal. There exist not one but several different definitions of the 2-D analytic signal that each zero out a different portion of the 2-D frequency plane [5]. We will consider two definitions. The first, proposed by Hahn in [6], employs complex algebra and zeros out frequencies on all but a single *quadrant* ($k_x, k_y > 0$, for example, where (k_x, k_y) indexes the 2-D frequency plane). In this formulation, the *complete* 2-D analytic signal consists of two parts: one having spectrum on the upper right quadrant ($k_x, k_y > 0$) and the other on the upper left quadrant ($k_x < 0, k_y > 0$) [5].

[1] A finitely supported function can never be exactly analytic [2]. In practice, we can only design finite-length complex wavelets that are approximately analytic, and thus the CWT is only approximately shift-invariant [3,4].

[2] Note that the Fourier transform of the complex scaling function, $\Psi_h(\omega) + j\Psi_g(\omega) = \Psi^c(\omega)$, is only approximately analytic in practice, and so its support will leak into the negative frequency region.

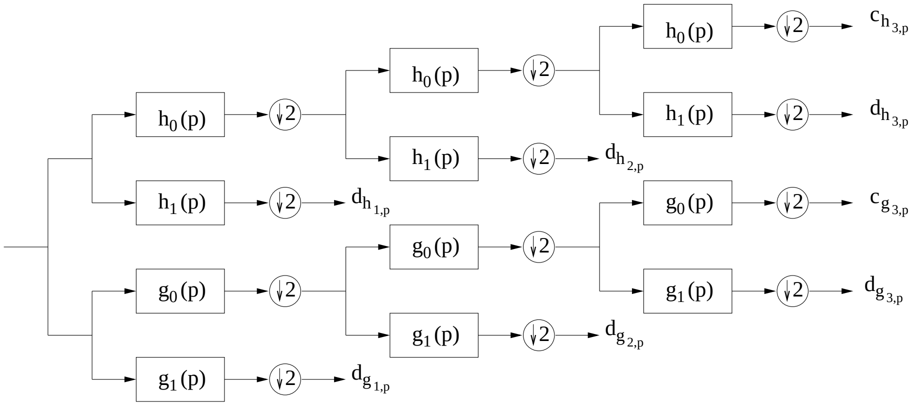


Figure 1: *The 1-D dual-tree CWT is implemented using a pair of filter banks operating on the same data simultaneously. Outputs of the filter banks are the dual-tree scaling coefficients, c_{h_p} and c_{g_p}, and the wavelet coefficients, $d_{h_{\ell,p}}$ and $d_{g_{\ell,p}}$, at scale ℓ and shift p. The CWT coefficients are then obtained as $d_{h_{\ell,p}} + j\, d_{g_{\ell,p}}$.*

Definition 1 [6] Let f be a real-valued, 2-D function. The *complete 2-D complex analytic signal* is defined in the space domain, $\mathbf{x} = (x, y)$, as the pair of complex signals

$$\begin{aligned} f_{A_1}(\mathbf{x}) &= [f(\mathbf{x}) - f_{\mathcal{H}i}(\mathbf{x})] + j[f_{\mathcal{H}i_1}(\mathbf{x}) + f_{\mathcal{H}i_2}(\mathbf{x})], && (2) \\ f_{A_2}(\mathbf{x}) &= [f(\mathbf{x}) + f_{\mathcal{H}i}(\mathbf{x})] + j[f_{\mathcal{H}i_1}(\mathbf{x}) - f_{\mathcal{H}i_2}(\mathbf{x})], && (3) \end{aligned}$$

where

$$\begin{aligned} f_{\mathcal{H}i_1}(\mathbf{x}) &= f(\mathbf{x}) ** \frac{\delta(y)}{\pi x}, && (4) \\ f_{\mathcal{H}i_2}(\mathbf{x}) &= f(\mathbf{x}) ** \frac{\delta(x)}{\pi y}, && (5) \\ f_{\mathcal{H}i}(\mathbf{x}) &= f(\mathbf{x}) ** \frac{1}{\pi^2 xy}. && (6) \end{aligned}$$

The function $f_{\mathcal{H}i}$ is the *total HT*; the functions $f_{\mathcal{H}i_1}$ and $f_{\mathcal{H}i_2}$ are the *partial HTs*; $\delta(x)$ and $\delta(y)$ are impulse sheets along the y-axis and x-axis respectively; and $**$ denotes 2-D convolution.

The 2-D complex analytic signal in (2)–(3) is the notion behind the 2-D dual-tree CWT [3,4]. Each 2-D CWT basis function is a 2-D complex analytic signal consisting of a standard DWT tensor wavelet plus three additional real wavelets obtained by 1-D HTs along either or both coordinates. For example, starting from real DWT's diagonal-subband tensor product wavelet $f(\mathbf{x}) = \psi_h(x)\psi_h(y)$ from above we obtain from equations (4)–(6) its partial and total HTs

$$(f_{\mathcal{H}i_1}, f_{\mathcal{H}i_2}, f_{\mathcal{H}i}) = (\psi_g(x)\psi_h(y), \psi_h(x)\psi_g(y), \psi_g(x)\psi_g(y)).$$

From Definition 1, we then obtain the two complex wavelets

$$\begin{aligned} \psi_1^c(x, y) &= (\psi_h(x)\psi_h(y) - \psi_g(x)\psi_g(y)) + j(\psi_h(x)\psi_g(y) + \psi_g(x)\psi_h(y)), && (7) \\ \psi_2^c(x, y) &= (\psi_h(x)\psi_h(y) + \psi_g(x)\psi_g(y)) + j(\psi_h(x)\psi_g(y) - \psi_g(x)\psi_h(y)), && (8) \end{aligned}$$

having orientations, $45°$ and $-45°$, respectively. Similar expressions can be obtained for the other two subbands ($\pm 15°$ and $\pm 75°$) based on $\psi_h(x)\phi_h(y)$ and $\phi_h(x)\psi_h(y)$.

Each 2-D CWT coefficient has only a single phase angle, which encodes the 1-D shift of image features perpendicular to its subband direction. Figure 2(a) illustrates this phase-shift property. This encoding may be sufficient for local 1-D structures such as edges, since we can define edge shifts uniquely by a single value, say r, in the direction perpendicular to the edge. However, even in this case, the analysis is not so straightforward when the edge does not align with the six orientations of the CWT subbands. Moreover, shifts of intrinsically 2-D (non-edge) image features such as in Figure 2(a) require two values (r_1, r_2) in the x and y directions, respectively. This creates ambiguity in the CWT phase shift. We can resolve this ambiguity by using the coefficients from two CWT subbands, but this complicates the use of the CWT for image analysis, modeling, and other image processing applications. In contrast, Figure 2(b) illustrates a more convenient encoding of image shifts in absolute x, y-coordinates (with two phase angles) using the quaternion phases of our new QWT, to which we now turn our attention.

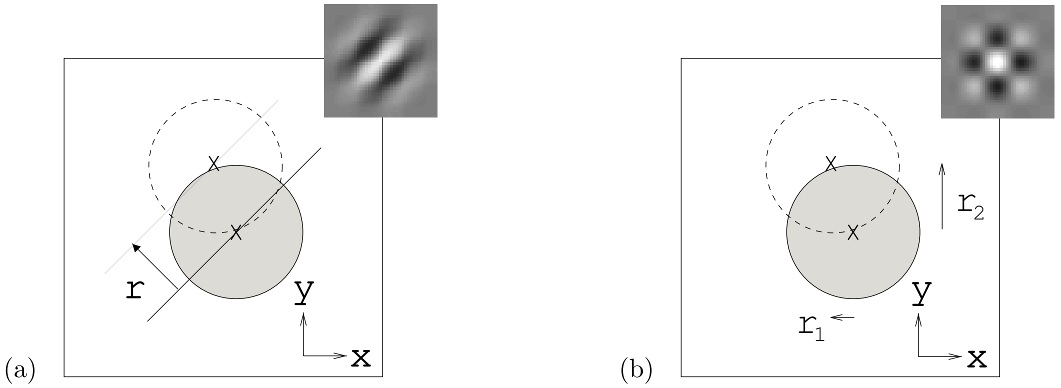


Figure 2: *(a) The CWT coefficient's single phase angle responds linearly to image shift r in a direction orthogonal to the wavelet's orientation. (b) Two of the QWT coefficient's three phase angles respond linearly to image shifts (r_1, r_2) in an absolute horizontal/vertical coordinate system.*

2.3 The Quaternion Wavelet Transform

2.3.1 Quaternion Hilbert transform

There are several alternatives to the 2-D analytic signal of Definition 1; we focus here on one due to Bülow [5]. It combines the partial and total HTs from (4)–(6) to form an analytic signal comprising a real part and three imaginary components that are manipulated using *quaternion* algebra [7].

The set of quaternions $\mathbb{H} = \{a + j_1 b + j_2 c + j_3 d \,|\, a, b, c, d \in \mathbb{R}\}$ has multiplication rules $j_1 j_2 = -j_2 j_1 = j_3$ and $j_1^2 = j_2^2 = -1$ as well as component-wise addition and multiplication by real numbers [8]. Additional multiplication rules include: $j_3^2 = -1$, $j_2 j_3 = -j_3 j_2 = j_1$ and $j_3 j_1 = -j_1 j_3 = j_2$. Note that quaternionic multiplication is not commutative. The conjugate q^* of a quaternion $q = a + j_1 b + j_2 c + j_3 d$ is defined by $q^* = a - j_1 b - j_2 c - j_3 d$ while the magnitude is defined as $|q| = \sqrt{qq^*}$.

An alternative representation for a quaternion is through its magnitude and three phase angles: $q = |q|\, e^{j_1\theta_1} e^{j_3\theta_3} e^{j_2\theta_2}$ [7], where $(\theta_1, \theta_2, \theta_3)$ are the quaternion phase angles, computed using the following formulae (for q normalized, i.e., $|q| = 1$):

$$\theta_3 = -\frac{1}{2}\arcsin(2(bc - ad))\,, \tag{9}$$

and, in the regular case (i.e., when $\theta_3 \in (-\frac{\pi}{4}, \frac{\pi}{4})$),

$$\theta_1 = \frac{1}{2}\arctan\left(\frac{2(bd + ac)}{a^2 + b^2 - c^2 - d^2}\right), \tag{10}$$

$$\theta_2 = \frac{1}{2}\arctan\left(\frac{2(cd + ab)}{a^2 - b^2 + c^2 - d^2}\right). \tag{11}$$

In the *singular case*, i.e., when $\theta_3 = \pm\frac{\pi}{4}$, θ_1 and θ_2 are not uniquely defined. Only the sum (if $\theta_3 = -\frac{\pi}{4}$) or the difference (if $\theta_3 = \frac{\pi}{4}$) of θ_1 and θ_2 is unique [7]. If $(\theta_1,\theta_2,\theta_3)$ calculated from (9)–(11) satisfy $e^{j_1\theta_1} e^{j_3\theta_3} e^{j_2\theta_2} = -q$, subtract θ_3 by π if $\theta_3 \geq 0$; add π to θ_3 if $\theta_3 < 0$. As a result, each quaternion phase angle is uniquely defined within the range $(\theta_1, \theta_2, \theta_3) \in [-\pi, \pi) \times [-\frac{\pi}{2}, \frac{\pi}{2}) \times [-\frac{\pi}{4}, \frac{\pi}{4}]$.

The operation of conjugation in the usual set of complex numbers, $\mathbb{C} = a + jb$, where $a, b \in \mathbb{R}$ and $j^2 = -1$, is a so-called *algebra involution* that fulfills the two following properties for any $z, w \in \mathbb{C}$: $(z^*)^* = z$ and $(wz)^* = w^* z^*$. In $\mathbb{H}$, there are three nontrivial algebra involutions:

$$\alpha_1(q) = -j_1 q j_1 = a + j_1 b - j_2 c - j_3 d, \tag{12}$$

$$\alpha_2(q) = -j_2 q j_2 = a - j_1 b + j_2 c - j_3 d, \tag{13}$$

$$\alpha_3(q) = -j_3 q j_3 = a - j_1 b - j_2 c + j_3 d. \tag{14}$$

Using these involutions we can extend the definition of Hermitian symmetry. A function $f : \mathbb{R}^2 \to \mathbb{H}$ is called *quaternionic Hermitian* if, for each $(x, y) \in \mathbb{R}^2$,

$$f(-x, y) = \alpha_2(f(x, y)), f(x, -y) = \alpha_1(f(x, y)) \text{ and } f(-x, -y) = \alpha_3(f(x, y)). \tag{15}$$

Bülow introduces an alternative definition of 2-D analytic signal based on the *quaternion Fourier transform* (QFT) [7]. The QFT of a 2-D signal f is given by

$$\mathcal{F}^q\{f\} = F^q(\mathbf{u}) = \int_{\mathbb{R}^2} e^{-j_1 2\pi ux} f(\mathbf{x})\, e^{-j_2 2\pi vy} d\mathbf{x}, \tag{16}$$

where $\mathcal{F}^q$ denotes the QFT operator, $\mathbf{u} = (u, v)$ indexes the QFT domain and the quaternion exponential

$$e^{-j_1 2\pi ux}\, e^{-j_2 2\pi vy} \tag{17}$$

is the QFT basis function. The real part of (17) is $\cos(2\pi ux)\cos(2\pi vy)$, while the other three quaternionic components are its partial and total HTs as defined in (4)–(6). Note that the QFT of a real-valued signal is quaternionic Hermitian, and each QFT basis function satisfies the definition of a 2-D *quaternion analytic signal.*

Definition 2 [5] Let f be a real-valued 2-D signal. The 2-D *quaternion analytic signal* is defined as

$$f_A^q(\mathbf{x}) = f(\mathbf{x}) + j_1 f_{\mathcal{H}i_1}(\mathbf{x}) + j_2 f_{\mathcal{H}i_2}(\mathbf{x}) + j_3 f_{\mathcal{H}i}(\mathbf{x}), \tag{18}$$

where the functions $f_{\mathcal{H}i_1}$, $f_{\mathcal{H}i_2}$ and $f_{\mathcal{H}i}$ are defined as in (4)–(6).

2.3.2 QWT construction

Our new 2-D dual-tree QWT rests on the quaternion definition of 2-D analytic signal. By organizing the four quadrature components of a 2-D wavelet (the real wavelet and its 2-D HTs) as a quaternion, we obtain a 2-D analytic wavelet and its associated *quaternion wavelet transform* (QWT). For example, for the diagonal subband, with $(f, f_{\mathcal{H}i_1}, f_{\mathcal{H}i_2}, f_{\mathcal{H}i}) = (\psi_h(x)\psi_h(y), \psi_g(x)\psi_h(y), \psi_h(x)\psi_g(y), \psi_g(x)\psi_g(y))$, we obtain the quaternion wavelet

$$\psi^{\mathrm{D}}(x, y) = \psi_h(x)\psi_h(y) + j_1\psi_g(x)\psi_h(y) + j_2\psi_h(x)\psi_g(y) + j_3\psi_g(x)\psi_g(y). \tag{19}$$

To compute the QWT coefficients, we can use a separable 2-D implementation [4] of the dual-tree filter bank in Figure 1. During each stage of filtering, we independently apply the two sets of h and g wavelet filters to each dimension (x and y) of a 2-D image; for instance, applying the set of filters h to both dimensions yields the scaling coefficients $c_{hh_{L,p}}$ and the diagonal, vertical, and horizontal wavelet coefficients, $d^{\mathrm{D}}_{hh_{\ell,p}}$, $d^{\mathrm{V}}_{hh_{\ell,p}}$, and $d^{\mathrm{H}}_{hh_{\ell,p}}$ respectively. Therefore, the resulting 2-D dual-tree implementation comprises four independent filter banks (corresponding to all possible combinations of wavelet filters applied to each dimension (hh, hg, gh, and gg)) operating on the same 2-D image. We combine the wavelet coefficients of the same subband from the output of each filter bank using quaternion algebra to obtain the QWT coefficients; for example, for the diagonal subband: $d^{\mathrm{D}}_{\ell,p} = d^{\mathrm{D}}_{hh_{\ell,p}} + j_1 d^{\mathrm{D}}_{gh_{\ell,p}} + j_2 d^{\mathrm{D}}_{hg_{\ell,p}} + j_3 d^{\mathrm{D}}_{gg_{\ell,p}}$.

Figure 3(c) illustrates the four components of a quaternion wavelet and its quaternion magnitude for the diagonal subband. The partial and total HT components resemble $\psi_h(x)\psi_h(y)$ but are phase-shifted by 90° in the horizontal, vertical, and both directions. The magnitude of each quaternion wavelet (square root of the sum-of-squares of all four components) is a smooth bell-shaped function. We can also interpret the four components of $\psi^{\mathrm{D}}(x, y)$ in the QFT domain as multiplying the quadrants of the QFT of $\psi_h(x)\psi_h(y)$ by $\pm j_1$ or $\pm j_2$, or both, as shown in Figure 4. Note that the order of multiplication is important because quaternion multiplication is non-commutative. This quaternion wavelet, $\psi^{\mathrm{D}}(x, y)$, has support in only a single quadrant of the QFT domain .

The construction and properties are similar for the other two subband quaternion wavelets based on $\phi_h(x)\psi_h(y)$ and $\psi_h(x)\phi_h(y)$ (see the horizontal and vertical subband wavelets, $\psi^{\mathrm{H}}(x, y)$ and $\psi^{\mathrm{V}}(x, y)$ in Figures 3(a) and (b), respectively). In summary, in contrast with the six complex pairs of CWT wavelets (12 functions in total), the QWT sports three quaternion sets of four QWT wavelets (12 functions in total).

Finally, note that the quaternion wavelet transform is approximately a windowed quaternion Fourier transform (QFT) [7]. In contrast to the QFT in (16), the basis functions for the QWT are scaled and shifted versions of the quaternion wavelets $(\psi_h(x) + j_1\psi_g(x))(\phi_h(y) + j_2\phi_g(y))$, $(\phi_h(x) + j_1\phi_g(x))(\psi_h(y) + j_2\psi_g(y))$, and $(\psi_h(x) + j_1\psi_g(x))(\psi_h(y) + j_2\psi_g(y))$.

2.4 QWT Applications

In this section, we demonstrate the utility of the QWT with two applications in geometrical edge structure and image disparity estimation.

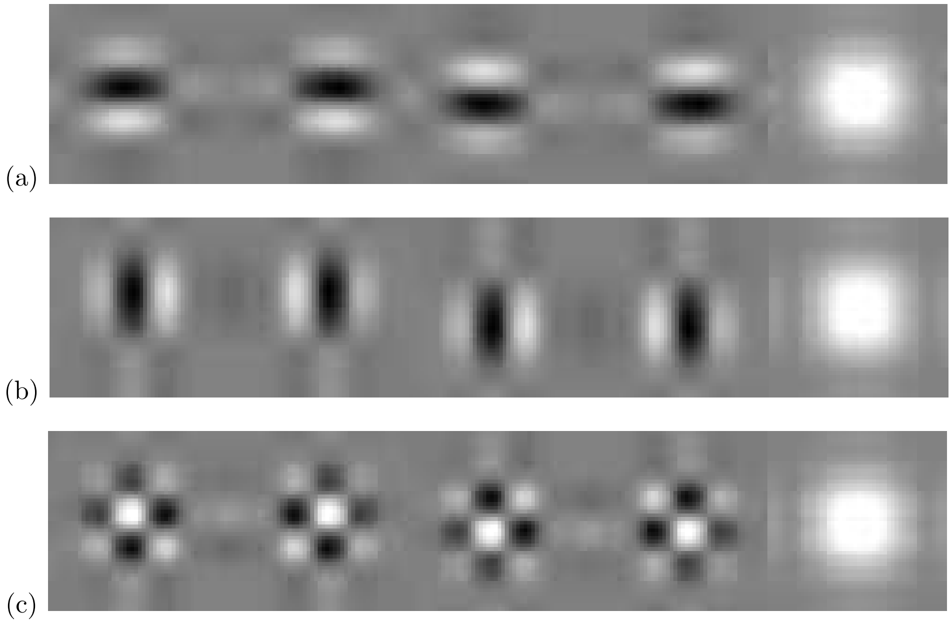


Figure 3: *Three quaternion wavelets from the 2-D dual-tree QWT frame. Each quaternion wavelet comprises four components that are 90° phase shifts of each other in the vertical, horizontal, and both directions. (a) Horizontal subband, from left to right: $\phi_h(x)\psi_h(y)$ (a usual, real DWT tensor wavelet), $\phi_h(x)\psi_g(y)$, $\phi_g(x)\psi_h(y)$, $\phi_g(x)\psi_g(y)$,$|\psi^{\mathrm{H}}(x,y)|$. (b) Vertical subband, from left to right: $\psi_h(x)\phi_h(y)$ (a usual, real DWT tensor wavelet), $\psi_h(x)\phi_g(y)$, $\psi_g(x)\phi_h(y)$, $\psi_g(x)\phi_g(y)$,$|\psi^{\mathrm{V}}(x,y)|$. (c) Diagonal subband, from left to right: $\psi_h(x)\psi_h(y)$ (a usual, real DWT tensor wavelet), $\psi_h(x)\psi_g(y)$, $\psi_g(x)\psi_h(y)$, $\psi_g(x)\psi_g(y)$, $|\psi^{\mathrm{D}}(x,y)|$. The image on the far right is the quaternion wavelet magnitude for each subband, a non-oscillating function.*

2.4.1 Edge geometry estimation

Edges are the fundamental building blocks of many real-world images. Roughly speaking, a typical natural image can be considered as a piecewise smooth signal in 2-D that contains blocks of smooth regions separated by edge discontinuities due to object occlusions. Here we use the QWT magnitude and phase to extend the multiscale edge geometry analysis of [9, 10].

Theory

Consider an image $f(\mathbf{x})$ containing a dyadic image block that features a single step edge, as parameterized in Figure 5(a). Note that for an edge oriented at angle β, any shift (r_1, r_2) in the (x, y) directions satisfying the constraint

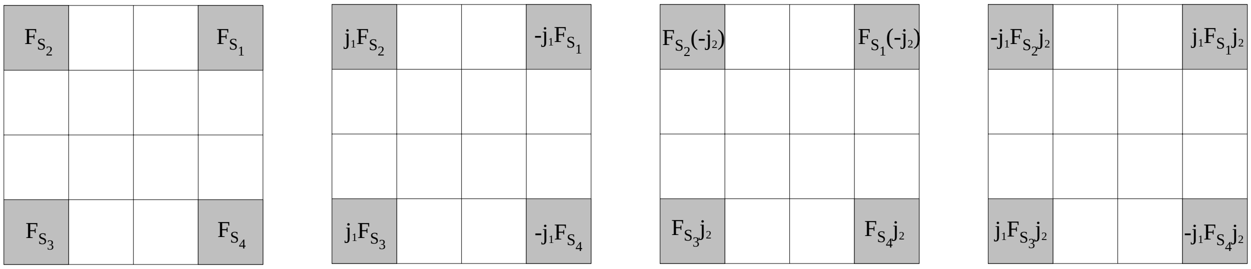
$$r_1 \cos\beta + r_2 \sin\beta = r \tag{20}$$

is identical to a shift from the center of the block by r in the direction perpendicular to β.

Our goal in this section is to analyze the phase $(\theta_1, \theta_2, \theta_3)$ of the quaternion wavelet coefficient (e.g., $d^{\mathrm{V}}_{\ell,p}$ for the vertical subband) corresponding to the quaternion wavelet $\psi^{\mathrm{V}}_{\ell,p}(\mathbf{x})$ whose support aligns with the dyadic image block containing the edge (the other subbands behave similarly). We will show that (θ_1,θ_2) and θ_3 provide an accurate means with which to estimate the edge offset r and edge orientation β respectively.

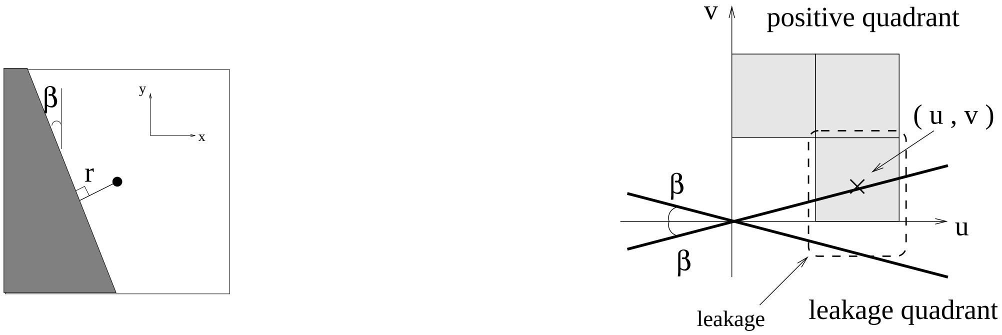
First, we establish a linear relationship between (θ_1,θ_2) and the edge offset r using the shift theorem of [12]. The effective center frequency (u, v) depends on both the image QFT spectral content $F^q(\mathbf{u})$ and the center frequency of the quaternion wavelet, and it always lies in the first quadrant. Since the spectral energy of the edge QFT, $F^q(\mathbf{u})$, concentrates along two impulse ridges through the origin having orientations $90^\circ - \beta$ and $\beta - 90^\circ$ in the QFT domain (see Figure 5(b) and the appendix of [12]), we can write $(u, v) = c(|\cos\beta|, |\sin\beta|)$, where c is a positive constant that depends on β, the subband, and the scale of analysis ℓ. When the edge passing through the image block center displaces perpendicularly by r, the changes in phase angles $(\Delta\theta_1,\Delta\theta_2)$ satisfy $r_1 = \frac{\Delta\theta_1}{2\pi u}$ and $r_2 = \frac{\Delta\theta_2}{2\pi v}$. Plugging $(u, v) = c(|\cos\beta|, |\sin\beta|)$ and using (20), we obtain the concise formula

$$r = \frac{\Delta\theta_1 \pm \Delta\theta_2}{2\pi c}, \tag{21}$$



(a) $\mathcal{F}^q\{\psi_h(x)\psi_h(y)\}$ (b) $\mathcal{F}^q\{\psi_g(x)\psi_h(y)\}$ (c) $\mathcal{F}^q\{\psi_h(x)\psi_g(y)\}$ (d) $\mathcal{F}^q\{\psi_g(x)\psi_g(y)\}$

Figure 4: *Quaternion Fourier domain relationships among the four quadrature components of a quaternion wavelet $\psi^D(x,y)$ in the diagonal subband. The QFT spectra of the real wavelet $\psi_h(x)\psi_h(y)$ in the first to fourth quadrants are denoted by $(F_{S_1}, F_{S_2}, F_{S_3}, F_{S_4})$ respectively. The partial and total Hilbert transform operations are equivalent to multiplying the quadrants of $\mathcal{F}^q\{\psi_h(x)\psi_h(y)\}$ in (a) by $\pm j_1$, or $\pm j_2$, or both.*



(a) single edge model (wedgelet) (b) edge QFT spectrum

Figure 5: *(a) Parameterization of a single edge in a dyadic image block (a wedgelet [11]). (b) QFT spectrum of the edge; shaded squares represent the quaternion wavelets in the horizonal, vertical, and diagonal subbands. The energy of the edge is concentrated along the two dark lines crossing at the origin and is captured by the vertical subband with effective center frequency at quaternion frequency (u, v). The region bounded by the dashed line demonstrates the spectral support of the QWT basis "leaking" into the neighboring quadrant.*

where we choose $\Delta\theta_1 + \Delta\theta_2$ when $\tan\beta > 0$, and $\Delta\theta_1 - \Delta\theta_2$ when $\tan\beta < 0$. We have verified this relationship via experimental analysis of straight edges in detail in earlier work [13].

Based on the interpretation of the QWT as a local QFT, we use an inner product formula to analyze the behavior of θ_3 for the same edge block. The QWT coefficient $d^{\mathrm{V}}_{\ell,p}$ can be computed from the QFT inner product between $\psi^{\mathrm{V}}_{\ell,p}(\mathbf{x})$ and the QFT of the edge image $f(\mathbf{x})$ (similarly for the other subbands). Our analysis in the appendix of [12] states that if the quaternion wavelet is perfectly analytic, then regardless of (β, r), $\theta_3 = \frac{\pi}{4}$ when $\tan\beta > 0$ and $\theta_3 = -\frac{\pi}{4}$ when $\tan\beta < 0$. Note that this corresponds to the singular case in the quaternion phase calculation in Section 2.3.1.

However, practical quaternion wavelets will not be perfectly analytic, and so their QFT support will leak into other quadrants of the QFT domain as in Figure 5(b). This necessitates the more in-depth analysis (see Appendix of [12]), which shows that in this case

$$\theta_3 = \frac{1}{2}\arcsin\left(\frac{1-\epsilon}{1+\epsilon}\right), \tag{22}$$

where ϵ is a measure of the ratio of local signal energy in the positive quadrant to the energy in the leakage quadrant. For the vertical subband as shown in Figure 5(b), when the edge orientation β changes from 0° to 45°, this ratio ϵ changes from 1 to 0 and thus θ_3 changes from 0 to $\frac{\pi}{4}$. We model this behavior of θ_3 in the horizontal and vertical subbands to design an edge orientation estimation in the next section. Since the diagonal subband wavelet has QFT support distant from the leakage quadrants, this QWT subband coefficients $d^{\mathrm{D}}_{\ell,p}$ are almost unaffected by leakage (i.e., $\epsilon \approx 0$). Their corresponding θ_3 approximately equal $\pm\frac{\pi}{4}$ and do not vary with β.

Practice

Based on the above analysis, we propose a hybrid algorithm to estimate the edge geometry (β, r) based on the QWT phase $(\theta_1, \theta_2, \theta_3)$ and the magnitude ratios between the three subbands. We generate a set of wedgelets with known β and r (see Figure 5(a) [11]) for analysis and testing. Our algorithm is reminiscent of the edge estimation scheme in [10].

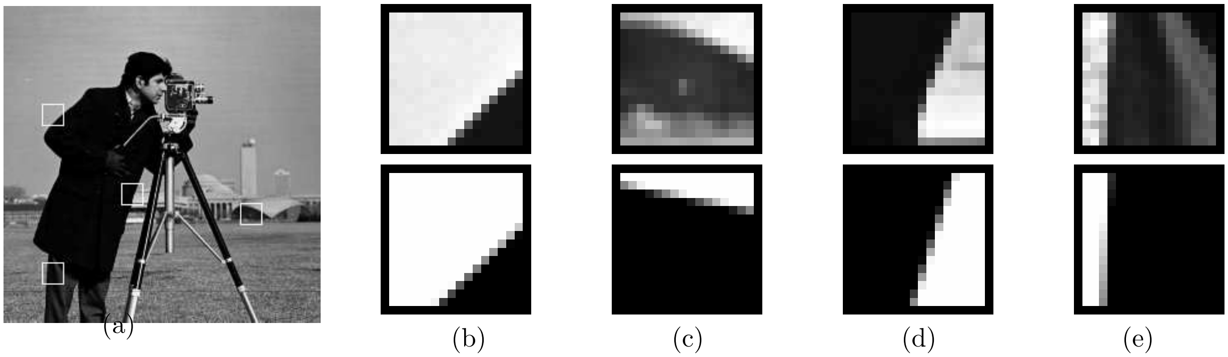


Figure 6: *Local edge geometry estimation using the QWT. (a) Several edgy regions from the "cameraman" image are shown on the left; (b)–(e) on the right are edge estimates from the corresponding QWT coefficients. The upper row shows the original image region, the lower row shows a wedgelet (see Figure 5(a)) having the edge parameter estimates (β, r). (No attempt is made to capture the texture within the block.)*

To estimate the edge orientation β, we use both the magnitude ratios among the three subbands and θ_3 of the subband with the largest magnitude. The subband with the largest magnitude gives the approximate orientation of the edge ($\pm 45°$ for diagonal, $\pm 15°$ for horizontal and $\pm 75°$ for vertical); the sign of θ_3 tells whether the direction is positive or negative. We experimentally analyze the QWT magnitude ratios and θ_3 of the set of generated wedgelets corresponding to changing edge orientations β by multiples of 5°. Using standard curve-fitting techniques, we develop a simple relationship between these parameters and β for our orientation estimation scheme. The resulting orientation estimation algorithm achieves a maximum error of only $\beta \pm 3°$ in practice for ideal edges.

To estimate the offset r of the edge, we apply the relationship between (θ_1,θ_2) and r in (21). We use (θ_1, θ_2) from either the horizontal or vertical subband (whichever has a larger magnitude). Depending on $\tan\beta$, we compute the sum (or difference) of the change in phase angles $\Delta\theta_1 \pm \Delta\theta_2$ for the edge under analysis, using $r = 0$ as the reference edge. Upon analysis of the simulated wedgelets with known r, we estimate $c \approx 0.7$, to be used in (21). Our final edge offset estimation algorithm achieves a maximum error of approximately ± 0.02 relative to the normalized unit edge length of the dyadic block (that is, sub-pixel accuracy). More details of the experimental analysis for the wedgelet model can be found in [13]. According to (21), within a 2π-range of $\Delta\theta_1 \pm \Delta\theta_2$, the range of r is limited to an interval of length $\frac{1}{c} \approx 1.43$ which ensures that the edge stays within the image block under analysis. Therefore, in our offset estimation, we need only consider one 2π-range of $\Delta\theta_1 \pm \Delta\theta_2$ and do not need to perform any "phase-unwrapping".

Finally, we estimate the *polarity* of the edge. By first obtaining an offset estimate for each *polarity* of the edge with orientation β estimated above, namely r^+ and r^-, we use the inner product between the image block and two wedgelets with the estimated edge parameters (β, r^+) and (β, r^-) to determine the correct polarity. Although our calculation of estimation accuracy is based on the wedgelet model, our algorithm also works well for real-world images such as the popular "cameraman" image in Figure 6.

Our results demonstrate the close relationship between edge geometry and QWT magnitude and phases, in particular, the encoding of edge location in the QWT phases (θ_1,θ_2) and the encoding of edge orientation in the QWT magnitude and third phase θ_3.

2.4.2 Image disparity estimation

In this section, as another example of QWT-based data processing, we present an algorithm to estimate the *local disparities* between the target image $A(x, y)$ and the reference image $B(x, y)$. *Disparity estimation* is the process of determining the local translations needed to align different regions in two images, that is, the amount of 2-D translation required to move a local region of a target image centered at pixel (x', y') to align with the region in a reference image centered at the same location (x', y'). This problem figures prominently in a range of image processing and computer vision tasks, such as video processing to estimate motion between successive frames, time-lapse seismic imaging to study changes in a reservoir over time, medical imaging to monitor a patient's body, super-resolution, etc.

Recall that the QWT phase property states that a shift (r_1, r_2) in an image changes the QWT phase from $(\theta_1, \theta_2, \theta_3)$ to $(\theta_1 - 2\pi u r_1, \theta_2 - 2\pi v r_2, \theta_3)$. Thus, for each QWT coefficient, if we know (u, v), the effective center frequency of the local image region analyzed by the corresponding QWT basis functions, then we can estimate the local image shifts (r_1, r_2) from the phase differences.

However, the center frequency (u, v) is image dependent in general. To be able to estimate image shifts from QWT phase differences, we first need to estimate (u, v) for each QWT coefficient. For this estimate, we can again use the QWT phase properties. If we know the image shifts and measure the phase difference, then we can compute (u, v).

By manually translating the reference image $A(x, y)$ by known small amounts both horizontally and vertically, we obtain two images $A(x, y)$ and $A(x - r_1, y - r_2)$. After computing the QWTs of $A(x, y)$ and $A(x - r_1, y - r_2)$, we can use the phase differences $(\Delta\theta_1, \Delta\theta_2)$ between the QWT coefficients to obtain estimates for the effective spectral center (u, v) for each dyadic block across all scales as $u = \frac{\Delta\theta_1}{2\pi r_1}$ and $v = \frac{\Delta\theta_2}{2\pi r_2}$. The range of QWT phase angles limits our estimates (u, v) to $\left[-\frac{1}{2R}, \frac{1}{2R}\right)$ and $\left[-\frac{1}{4R}, \frac{1}{4R}\right)$ for horizontal and vertical shifts, respectively, where R is the length of one side of the dyadic block corresponding to each coefficient.

Once we know the center frequency (u, v) for each QWT coefficient, we can estimate the local image shifts by measuring the difference between the QWT phase corresponding to the same local blocks in image $A(x, y)$ and $B(x, y)$.

A key challenge in phase-based disparity estimation is resolving the phase wrap-around effect due to the limited range of phase angles. Due to phase wrapping, each observed phase difference can be mapped to more than one disparity estimate. Specifically, for QWT phase differences $(\Delta\theta_1, \Delta\theta_2)$ between the reference and target images, we can express the possible image shifts of each dyadic block as

$$r_1 = \frac{\Delta\theta_1 + \pi(2n + k)}{2\pi u}, \qquad r_2 = \frac{\Delta\theta_2 + m\pi}{2\pi v}, \tag{23}$$

where $n, m \in \mathbb{Z}$ and $k \in \{0, 1\}$. Depending on m, k is chosen such that it equals 0 when m is even and equals 1 when m is odd. The special wrap-around effect in (23) is due to the limited range in θ_1 and θ_2 (to $[-\pi, \pi)$ and $[-\frac{\pi}{2}, \frac{\pi}{2})$ respectively).

In our multiscale disparity estimation technique, we use coarse scale shift estimates to help unwrap the phase in the finer scales. If we assume that the true image shift is small compared to the size of dyadic squares at the coarsest scale L, then we can set $m = n = k = 0$ in (23) at this scale (no phase wrap-around) and obtain correct estimates for r_1 and r_2. Effectively, this assumption of no phase wrap-around at the coarsest scale limits the maximum image shift that we can estimate correctly. Once we have shift estimates at scale L, for each block at scale $\ell = L - 1$, we estimate the shifts as follows:

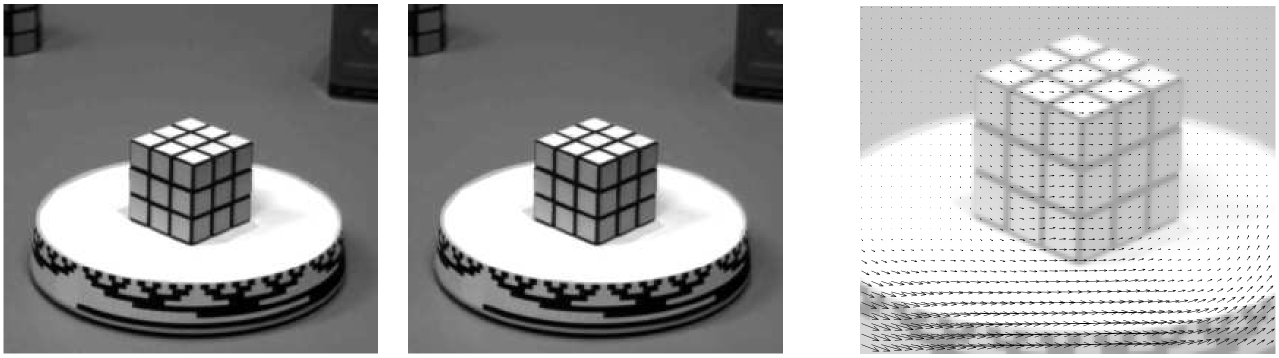
1. interpolate the estimates from the previous scale(s) to obtain predicted estimates (r_1^p, r_2^p),
2. substitute $(\Delta\theta_1, \Delta\theta_2)$ into (23) and determine the (n, k, m) such that (r_1, r_2) is closest to (r_1^p, r_2^p),
3. remove any unreliable (r_1, r_2),
4. repeat Steps 1–3 for the finer scales $\ell = L - 2, L - 3, \ldots$

Step 1 uses either nearest-neighbor interpolation (which gives higher estimationaccuracy) or bilinear interpolation (which results in a smoother disparity field for better visual quality). We choose the latter in our simulations. In Step 3, we use a similar reliability measure as in the confidence mask [7] to threshold unreliable phase and offset estimates. We also threshold based on the magnitude of the QWT coefficients. We iterate the above process until a fine enough scale (e.g., $\ell = 2$), since estimates typically become unreliable at this scale and below. The QWT coefficients for the small dyadic blocks have small magnitudes, and so their phase angles are very sensitive to noise.

We can improve upon the basic iterative algorithm by fusing estimates across subbands and scales. First, with proper interpolation, we can average over estimates from all scales containing the same image block. Second, we can average estimates from the three QWT subbands for the same block to yield more accurate estimates, but we need to discard some unreliable subband estimates (for example, horizontal disparity r_1 in the horizontal subband and r_2 in the vertical subband). We incorporate these subband/scale averaging steps into each iteration of Steps 1–4.[3]

Figure 7 illustrates the result of our QWT phase-based disparity estimation scheme for two frames from the rotating Rubik's cube video sequence [7]. This is an interesting sequence, because a rotation cannot be captured by a single global translation but can be closely approximated by local translations. The arrows indicate both the directions and magnitudes of the local shifts, with the magnitudes stretched for better visibility. We can clearly see the rotation of the Rubik's cube on the circular stage, with larger translations closer to the viewer (bottom of the image) and smaller translations further from the viewer (top of the image). In our experiments, we obtained the most robust estimations by averaging over both scales and subbands.

[3] Matlab software is available at http://www.dsp.rice.edu/software/qwt.shtml.

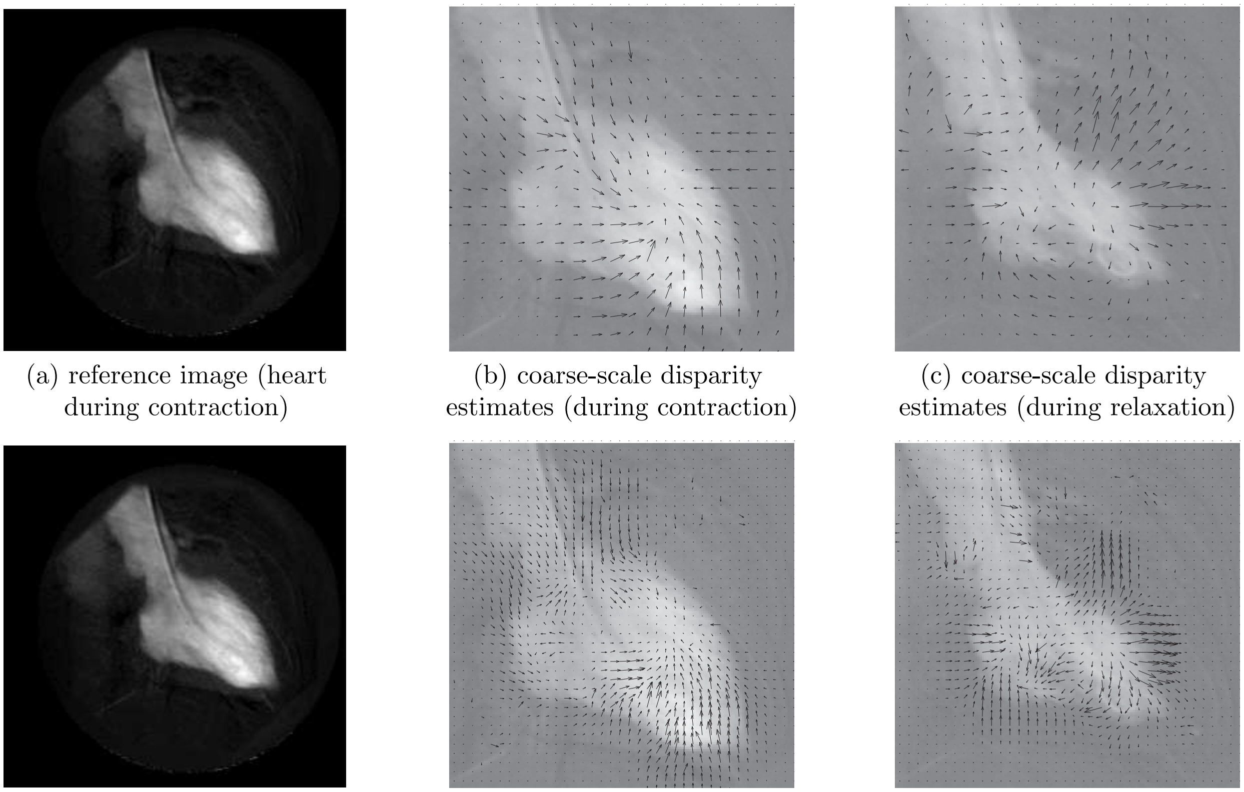


(a) reference image

(b) target image

(c) disparity estimates

Figure 7: *Multiscale QWT phase-based disparity estimation results. (a), (b) Reference and target images from the "Rubik's cube" image sequence [7]. (c) Disparity estimates between two images in the sequence, shown as arrows overlaid on top of the reference image (zoomed in for better visualization, arrow lengths proportional to magnitude of disparity).*



(a) reference image (heart during contraction)

(b) coarse-scale disparity estimates (during contraction)

(c) coarse-scale disparity estimates (during relaxation)

(d) target image (heart during contraction)

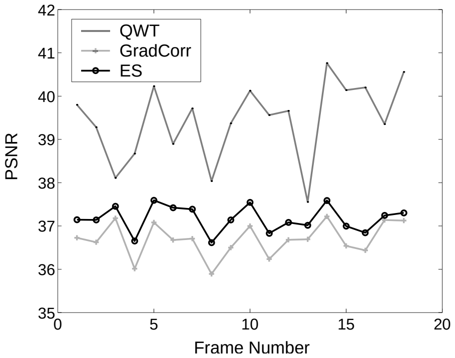
(e) fine-scale disparity estimates (during contraction)

(f) fine-scale disparity estimates (during relaxation)

Figure 8: *Multiscale QWT phase-based disparity estimation results for the "heart" image sequence. (a), (d) Reference and target "heart "images from two frames during heart contraction (systole). Estimation results show (b)-(c) coarse-scale and (e)-(f) fine-scale detailed motion of the heart and blood flow during the contraction and expansion (systole and diastole) phases of the cardiac cycle, illustrating the multiscale nature of our algorithm.*

Figure 8 demonstrates the multiscale nature of our disparity estimation approach on an image sequence of a living heart [14]. The presence of sharp edges plus the smooth cardiac motion in these images is well-matched by our wavelet-based analysis. Our algorithm averages over several scales ℓ to obtain motion fields in various levels of detail. Figure 8(b) and (c) clearly show the coarse-scale contraction and relaxation motions of the heart, while Figure 8(e) and (f) display more detailed movements of the heart muscles, arteries and blood flow (in particular, see the arrows toward left region of the heart near the artery).

In addition to visualizing the changes from one image to another, we can use our algorithm as the feature



(a) PSNR vs. frame number

	QWT	GC	ES
Rubik	**39.4**	36.7	37.2
Heart	35.7	35.5	**38.2**
Taxi	36.2	36.6	**37.0**
Computation cost	$O(N)$	$O(N\log N)$	$O(N^2)$

(b) average PSNR performance (in dB)

Figure 9: *Comparison of multiscale QWT phase-based disparity estimation with two motion estimation algorithms, Gradient Correlation (GC) [15] and exhaustive search (ES). The performance measure is PSNR (in dB) between the motion-compensated image and the target image of four test image sequences ("Rubik", "Heart" and "Taxi"). (a) Frame-by-frame PSNR performance comparison in the "Rubik" sequence. (b) Table of average PSNR performance (over all frames) for each test sequence. The multiscale QWT phase-based method demonstrates the best performance among the three test algorithms for the "Rubik" sequence and shows comparable performance to the other algorithms for the "Heart" and "Taxi" sequences. Last row of table shows the computational complexity of each algorithm.*

matching step of an *image registration* algorithm in the process, using the disparity information to build a warping function to align the images. One important note is that our QWT-based method is *region-based* in that it does not require any detection of feature points or other image landmarks. Traditional region-based feature matching methods, which estimate the spatial correlation (or correspondence) between two images, include *block-matching* and Fourier methods. For comparison, we compare our approach to the *exhaustive search* (ES) block matching technique, which is very computationally demanding but features the best performance among all general block-matching techniques. We also compare to a Fourier sub-pixel motion estimation method known as *Gradient Correlation* (GC), which has been shown to have better PSNR performance than other recent Fourier methods [15].

As a performance measure, we use the Peak Signal-to-Noise Ratio (PSNR) between the motion compensated image $C(x,y)$ and the target image $B(x,y)$, which is given by

$$10\log_{10}\left(\frac{(255)^2 N}{\sum_{\mathbf{x}}(B(\mathbf{x})-C(\mathbf{x}))2)}\right), \tag{24}$$

where N is the number of image pixels. The motion compensated image $C(x,y)$ is obtained by shifting each image block in the reference image $A(x,y)$ according to the estimated motion vectors. Figure 9 compares the results for four image sequences: the "Rubik" and "Taxi" sequences commonly used in the optical flow literature, and the "Heart" sequence from Figure 8. Figure 9(a) demonstrates the superior performance of our QWT phase-based algorithm over the other algorithms for the "Rubik" sequence, which has piecewise-smooth image frames and a smooth underlying disparity flow. While its PSNR performance is relatively far from ES for the "Heart" sequence, we note that the QWT phase-based approach provides a motion field that is more useful for patient monitoring and diagnosis. For the "Taxi" and "Mobcal" sequences, which contain discontinuities in their underlying flows, the QWT phase-based algorithm sports comparable performance (see the table in Figure 9(b)). Since the multiscale averaging step in our algorithm tends to smooth out the estimated flow, it should not be expected to perform as well for discontinuous motions fields of rigid objects moving past each other.

Additional advantages of our QWT-based algorithm include its speed (linear computational complexity) and sub-pixel estimation accuracy. For an N-pixel image, our $O(N)$ algorithm is more efficient than the $O(N\log N)$ FFT-based GC and significantly faster than ES, which can take up to $O(N^2)$ with the search parameter on the order of N. General block-matching techniques such as ES can only decipher disparities in an integer number of pixels. On the other hand, our QWT-based algorithm can achieve sub-pixel estimation and demonstrates greater accuracy for the "Rubik" sequence than existing phase-based sub-pixel estimation methods such as GC.

Besides gradient correlation, there exist other phase-based algorithms for disparity estimation and image registration [7, 16–20]. These approaches use phase as a feature map $\Phi(\mathbf{x})$ where the phase function Φ maps 2-D x, y-coordinates to phase angles. They assume the phase function to stay constant upon a shift from the reference image to the target image; that is, $\Phi_1(\mathbf{x}) = \Phi_2(\mathbf{x}+\mathbf{r})$ where Φ_1 is the phase function for the reference image and Φ_2 for the target image. Then, the disparity estimation problem is simplified to calculating the optical flow for these phase functions [16,21]. In contrast, our algorithm is entirely based on the multiscale dual-tree QWT and its shift theorem.

Our approach is similar to Bayro-Corrochano's QWT disparity estimation algorithm in [19] in its use of quaternion phase angles. However, the latter approach requires the design of a special filter to compute the phase derivative function in advance, while our approach need only estimate the local frequencies (u, v). Our implementation also uses a dual-tree filterbank, as compared to the quaternion wavelet pyramid of Gabor filters in [19]. Provided a *continuous* underlying disparity flow, our algorithm yields a denser and more accurate disparity map, even for smooth regions within an image. Incorporating an optimization procedure such as in [20] or a statistical model into our current algorithm can further improve estimation accuracy, particularly for blocks with phase singularity, but requires extra computation time.

Kingsbury et al. have developed a multiscale displacement estimation algorithm based on the 2-D CWT [16,17]. Their approach combines information from all six CWT subbands in an optimization framework based on the optical flow assumptions. In addition to disparity estimation, it simultaneously registers the target image to the reference image. In comparison, both the QWT and CWT methods are multiscale and wavelet-based and are thus, in general, best for smooth underlying disparity flows. However, our algorithm is much simpler and easier to use because it does not involve the tuning of many input parameters for the iterative optimization procedures as in the CWT algorithm. While our method estimates local disparities without warping the image, we can apply any standard warping procedure to easily register the two images from the estimated disparities. Thanks to the ability of the QWT to encode location information in absolute horizontal/vertical coordinates, we can easily combine the QWT subband estimates to yield more accurate flow estimation results. Combining subband location information in the 2-D CWT is more complicated, since each subband encodes the disparities by complex phase angles in a reference frame rotated from other subbands. Based on our experimental results and comparison of the design of our flow estimation algorithm with previous approaches, the QWT demonstrates its ability to efficiently represent, encode and process location information in images.

2.5 Conclusions and Future Work

We have introduced a new 2-D multiscale wavelet representation, the dual-tree QWT, that is particularly efficient for coherent processing of relative location information in images. This tight-frame transform generalizes complex wavelets to higher dimensions and inspires new processing and analysis methods for wavelet phase.

Our development of the dual-tree QWT is based on an alternative definition of the 2-D HT and 2-D analytic signal and on quaternion algebra. The resulting quaternion wavelets have three phase angles; two of them encode phase shifts in an absolute horizontal/vertical coordinate system while the third encodes textural information. The QWT's approximate shift theorem enables efficient and easy-to-use analysis of the phase behavior around edge regions. We have developed a novel multiscale phase-based disparity estimation scheme. Through efficient combination of disparity estimates across scale and wavelet subbands, our algorithm clearly demonstrates the advantages of coherent processing in this new QWT domain. Inherited from its complex counterpart, the QWT also features near shift-invariance and linear computational complexity through its dual-tree implementation.

Beyond 2-D, the generalization of the Hilbert transform to n-D signals using hypercomplex numbers can be used to develop higher dimensional wavelet transforms suitable for signals containing low-dimensional manifold structures [22]. The QWT developed here could play an interesting rôle in the analysis of $(n-2)$-D manifold singularities in n-D space. This efficient hypercomplex wavelet representation could bring us new ways to solve high-dimensional signal compression and processing problems.

3 The Geometric Hidden Markov Tree: A new geometrical statistical model

3.1 Summary of results

Traditional wavelet-based image processing algorithms and models have significant shortcomings in their treatment of edge contours. The standard modeling paradigm exploits the fact that wavelet coefficients representing smooth regions in images tend to have small magnitude, and that the multiscale nature of the wavelet transform implies that these small coefficients will persist across scale (the canonical example is the venerable zero-tree coder). The edge contours in the image, however, cause more and more large magnitude wavelet coefficients as we move down through scale to finer resolutions. But if the contours are smooth, they become simple as we zoom in on them, and are well approximated by straight lines at fine scales. Standard wavelet models exploit the grayscale regularity of the smooth regions of the image, but not the geometric regularity of the contours.

We have built a model that accounts for this geometric regularity by capturing the dependencies between complex wavelet coefficients along a contour. The Geometric Hidden Markov Tree (GHMT) assigns each wavelet coefficient (or spatial cluster of wavelet coefficients) a hidden state corresponding to a linear approximation of the local contour structure. The shift and rotational-invariance properties of the complex wavelet transform allow the GHMT to model the behavior of each coefficient given the presence of a linear edge at a specified orientation — the behavior of the wavelet coefficient given the state. By connecting the states together in a quadtree, the GHMT ties together wavelet coefficients along a contour, and also models how the contour itself behaves across scale.

3.2 Background and setup

Images have two salient features that any model should account for: they contain smooth, homogeneous regions, and these regions are separated by smooth edge contours. That is, images exhibit *grayscale regularity* in smooth regions and *geometric regularity* along edge contours. Both grayscale regular regions and geometric regular contours are readily characterized by their multiscale behavior. In a sense, they both become "uninteresting" as we zoom in on them; at fine resolutions, smooth regions are essentially flat and contours are essentially straight.

The *wavelet transform* handles the grayscale regularity in images naturally, and as a result has emerged as the preeminent tool in image processing. Wavelet models operate under the paradigm "wavelet coefficients representing smooth regions in the image tend to be small". This idea is the basis for many state-of-the-art wavelet domain processing algorithms for applications including compression [23, 24], denoising [25, 26], and segmentation [27].

An effective model for edge structure in the wavelet domain has been more elusive. At fine scales in the wavelet domain, many coefficients are needed to build up an edge (the number roughly doubles from scale to scale). But since the edge contour itself is simple (smooth), the values of the wavelet coefficients are "geometrically coherent" in that they exhibit strong dependencies, a fact which current models do not exploit.

To address this issue, recent research in applied harmonic analysis has focussed on developing alternate representations that are better suited for edges [28–32]. While these new representations have theoretically nice properties, they do not yet enjoy the same widespread success in practical applications as wavelets.

Instead of proposing an alternate representation, we explore a method of compensating for contour regularity directly in the wavelet domain. The approach combines three recent developments in image modeling: *complex wavelets* [33], *hidden Markov trees* [25], and *multiscale geometry modeling* [34].

The complex wavelet transform (CWT) [33,35], like the real wavelet transform, decomposes an image using basis functions that are local in time and frequency. Unlike the real wavelet transform, complex wavelets are also *local in orientation* (see Figure 10), approximately *shift-invariant* and approximately *steerable*. As a result, geometrical coherency among a group of complex wavelet coefficients is easy to identify and exploit.

In Section 3.4, we will develop the geometric hidden Markov tree (GHMT), a statistical model for images in the complex wavelet domain. The GHMT assigns a hidden state to each group of six complex wavelet coefficients that corresponds to the local linear edge structure present in the image (see Figure 14). The presence of an edge *constrains* the CWT coefficients to behave in a certain manner, just as they are "constrained" to be small when representing smooth regions.

The GHMT captures the contour structure in the image by imposing multiscale *dependencies* between these hidden states. Since each contour in the image is smooth, their local linear behavior is closely related across scale (see Figure 11). These relationships are quantified with transition probabilities between the hidden states of parent and child CWT coefficients on the wavelet quadtree. An appropriate choice of transition probabilities allows the GHMT to favor hidden state configurations that are *geometrically faithful*; at fine scales, the contour is essentially straight, and we expect little innovation in the hidden state sequence.

In Section 3.5, we apply the model to the problem of *feature extraction*. Given an image, we use a fast optimization algorithm to find the state configuration with greatest *joint* likelihood. The resulting hidden state configuration gives us an estimate of the contour structure in the image across a range of scales; an example is shown in Figure 15. The pictures in Figure 15 are completely parameterized; they were redrawn from the "vector graphics" information (line segment orientations) inherent in the state configuration.

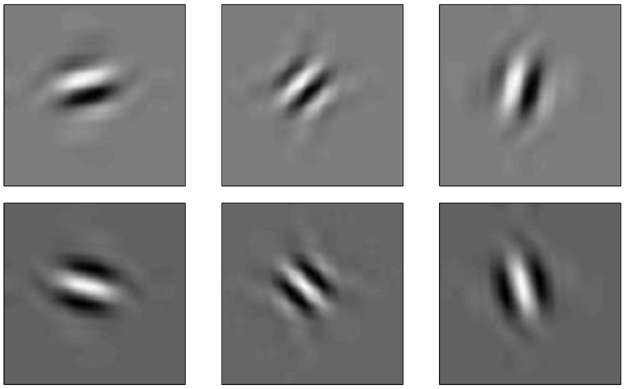


Figure 10: Real parts of the six 2D wavelet basis functions for a given scale and location. The basis functions are local in space, frequency, and direction; each responds only to edges at certain orientations.

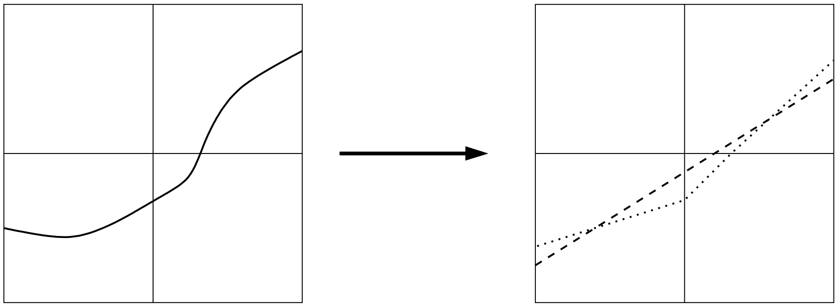


Figure 11: The linear behavior of the curve on the right is closely related across scale. The dashed line on the left shows the linear fit at one scale, and the dotted line shows the fits at the next finer scale.

3.3 Complex wavelets and local geometry

The complex wavelet transform decomposes an image $X(s), s \in \mathbb{R}^2$ in terms of basis functions that are shifted in dilated versions of *six* different mother wavelets ψ^b:

$$X(s) = \sum_{k \in \mathbb{Z}^2} u_{j_0,k} + \sum_{b \in \mathcal{B}} \sum_{j \geq j_0} \sum_{k \in \mathbb{Z}^2} c^b_{j,k} \psi^b_{j,k}(s) \tag{25}$$

where $\psi^b_{j,k}(s) = 2^j \psi^b(2^j s - k)$. The $c^b_{j,k}$ and $\psi^b_{j,k}(s)$ are complex valued, with the real and imaginary parts of $\psi^b_{j,k}$ forming an approximate Hilbert pair. The CWT uses six subbands $\mathcal{B} = \{15, 45, 75, -15, -45, -75\}$, labeled for the directional information they give. Like the real wavelet transform, each subband can be arranged on a quadtree with nodes indexed by scale and location. The real parts of the $\psi^b_{j,k}$ for one scale and location are shown in Figure 10.

The CWT has several properties which make it attuned to characterizing geometrical structure[4]:

Shift-invariance: Let $\{c^b_{j,k}\}_{b,k}$ be the CWT coefficients of an image X at a scale j. Then the corresponding scale j CWT coefficients for any *shift* of X can be calculated by interpolating the $\{c^b_{j,k}\}_{b,k}$.

Rotational-invariance: Similarly, the CWT coefficients for any *rotation* of X can be calculated by interpolating the $\{c^b_{j,k}\}_{b,k}$.

Directional selectivity: Along with being local in time and frequency, the $\psi^b_{j,k}$ are local in *orientation.*

As a result of these three properties, the CWT coefficients at one scale and location can provide a characterization of local geometrical structure. For a CWT basis function centered on location k in the plane, local edge positions can be parameterized by a distance d and an angle θ relative to k (see Figure 12). The (d, θ) parameterization provides a nice separation in the *edge response* of the six CWT coefficients (see Figure 13): d controls the *phase*, while θ controls the magnitude. The mapping from (d, θ) to the corresponding edge response is (almost) invertible.

[4] While not strictly true, theses properties hold for nearly all intents and purposes.

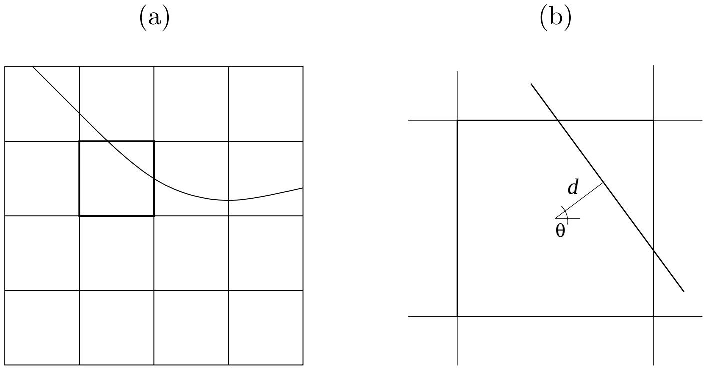


Figure 12: The local linear behavior of a contour inside the bold box in (a) is parameterized by (b) an angle θ and offset d.

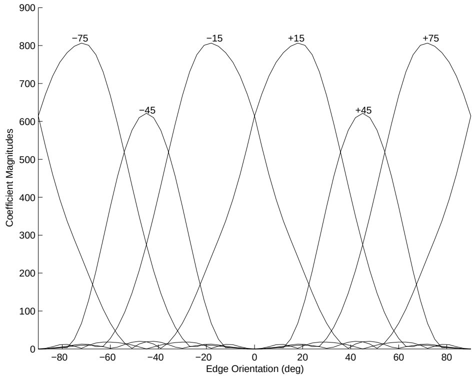


Figure 13: Magnitude edge response for the six CWT basis functions at a given scale and location as a function of θ for $d = 0$. The phase response varies linearly with d. The relative magnitudes are tied to the angle θ of local geometrical structure while the phase is tied to the offset d.

The local geometry is manifest in the CWT coefficients; simply by looking at their values, we can not only decide whether or not an edge is in the vicinity, but we can also estimate the distance d and angle θ. An example of local edge structure being "read off" from CWT coefficients is shown in Figure 6.

3.4 Geometric hidden Markov tree modeling

The wavelet-domain hidden Markov tree [25] (HMT) is a general purpose statistical image model underlying powerful denoising [26] and segmentation [27] algorithms. The HMT separates the wavelet coefficients $c^b_{j,k}$ of an image into two categories, S and L, and uses a different statistical model for each. Over smooth regions in the image, the wavelet coefficients are expected to have small magnitude; the HMT models these type S coefficients as coming from a low-variance Gaussian distribution. Over regions with edges or texture, the wavelet coefficients can have larger magnitude; these type L coefficients are modeled with a high-variance Gaussian. Of course, the type of each wavelet coefficient is not known a priori; it can be interpreted as a *hidden state* $q_{j,k}$ that controls the marginal distribution[5] of $c^b_{j,k}$

$$P(c^b_{j,k}|q_{j,k} = m) = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left(-c^b_{j,k}/(2\sigma^2_m)\right) \quad m \in \{\mathsf{S}, \mathsf{L}\}. \tag{26}$$

To model the dependencies between the wavelet coefficients, the HMT sets up a Markov-1 dependency structure on hidden states across scale. The relationships between parent and child coefficients[6] are parameterized by transition probability matrix

$$A_{mn} = \text{Prob}\left(\text{child state} = m|\text{parent state} = n\right). \tag{27}$$

[5] Typically, the variances σ^2_m will also vary with the scale j

[6] The wavelet coefficients of an image can be naturally arranged on a quadtree, where "parent nodes" at a scale j split into four "child" nodes at scale $j + 1$.

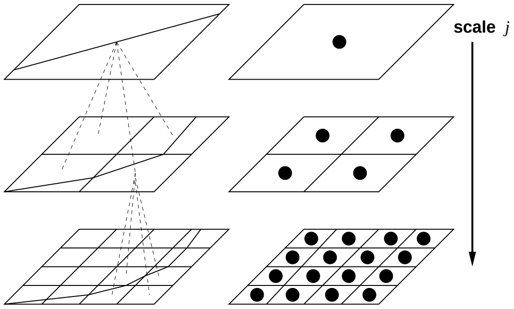


Figure 14: The geometric hidden Markov tree model. The tree of hidden states, shown on the left, models the multiscale behavior of the contours. Associated with each hidden state is a set of six complex wavelet coefficients (drawn as black dots on the right).

Since smooth regions of the image at one scale will break down into smooth regions at the next scale, we expect A_{SS} to be close to 1.

Rather than using a single "Gaussian with large variance" state to model contours in the image, the geometric HMT (GHMT) breaks the L state down into many *geometry states*, each corresponding to a different type of local linear contour behavior. The states are drawn from a dictionary of possible (d, θ) values, $q_{j,k} \in \ \mathsf{S} \cup \{(d_m, \theta_m)\}$, and we use the same state for all six CWT coefficients $\{\psi^b_{j,k}\}_b$ for a given scale and location (j, k).

As discussed in Section 3.3, each type of local linear contour structure produces distinct behavior in the CWT coefficients. Given a geometrical state $q_{j,k} = (d_m, \theta_m)$, we model the six CWT coefficients as coming from an improper Gaussian whose "mean" is the one dimensional subspace $E^{(d_m,\theta_m)}_{j,k}$ of response values of $\{\psi^b_{j,k}\}_b$ for edges of all different heights at (d_m, θ_m). We have

$$P\left(\{c^b_{j,k}\}_b | q_{j,k} = (d_m, \theta_m)\right) \propto \exp\left(-\operatorname{dist}(\{c^b_{j,k}\}_b, E^{(d_m,\theta_m)}_{j,k})^2/(2\sigma^2_g)\right) \tag{28}$$

where $\operatorname{dist}(\{c^b_{j,k}\}_b, E^{(d_m,\theta_m)}_{j,k})$ is the distance of the CWT coefficients $\{c^b_{j,k}\}_b$ to the subspace $E^{(d,\theta)}_{j,k}$ [7] and σ^2_g controls the variance around this subspace.

The transition probabilities will favor small innovations in geometry over large. To each state (d_m, θ_m), there corresponds a line ℓ_m (as in Figure 12(b)). The A_{mn} are assigned based on the Hausdorff distance $\mathrm{HD}(\ell_m, \ell_n)$ between lines ℓ_m and ℓ_n restricted to a square in the plane. A small Hausdorff distance corresponds to little change in geometry between parent and child, and results in a high probability. We use $A_{mn} \propto e^{-\mathrm{HD}(\ell_m,\ell_n)}$.

The GHMT, illustrated in Figure 14, is a general purpose statistical model for the complex wavelet coefficients of images, and thus can be applied to a range of image processing problems. In the next section, we present a simple feature extraction algorithm.

3.5 Application: Feature extraction

The hidden states in the GHMT are more than just a modeling device; they correspond to *semantic* contour information. Since edge contours are perceptually the most important features in an image, knowledge of the underlying states is interesting in its own right.

Given an image, the Markov-1 quadtree structure of the GHMT allows us to find the joint maximum-likelihood hidden state sequence $\{\widehat{q_{j,k}}\}$ using the Viterbi algorithm [36]. The result is a set of (d, θ) estimates at every node tied together by the geometry model. The collection of estimate states $\{\widehat{q_{j,k}}\}_k$ at a scale j can be used to "sketch" the contour structure in an image; the finer the scale, the more details are included.

An example of geometrical feature extraction using GHMT state estimation is shown in Figure 15. The contour sketches are re-created from the (d, θ) values inferred by the Viterbi algorithm. The estimated hidden state values are equivalent to a *vector graphics* representation of the contour sketch, and can be used to form an extremely low bitrate semantic approximation to the image.

3.6 Conclusions and Future Work

Despite their success, traditional wavelet-based models have significant shortcomings in their treatment of edge structure in images. We have introduced a modeling framework that accounts for geometrically smooth edge

[7] We could make the distribution in (28) proper by putting a prior on the *heights* of the edges in the image.

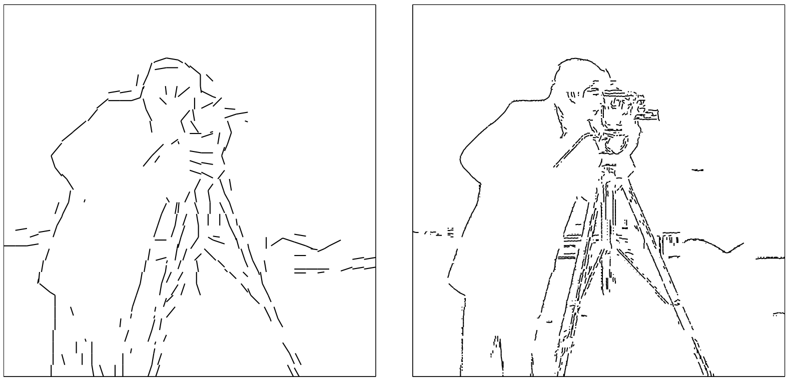


Figure 15: Contour sketches at two different scales of the "cameraman" image shown in Figure 6. The sketches were created directly from the (d, θ) values of the geometrical states estimated by the Viterbi algorithm.

structure. The geometrical HMT models spatial clusters of complex wavelet coefficients according to an underlying hidden state specifying a linear approximation of the local contour structure. By imposing dependencies on these hidden states, we are able incorporate a characterization of geometrical structure into the model. Future areas of work include the application of the model to other signal processing applications, such as compression and denoising. Another fruitful area of work lies in the combination of this model with other geometric models, and the pursuit of new harmonic bases motivated by geometry.

4 Surflet representation and compression of higher dimensional functions containing smooth discontinuities

4.1 Summary of results

Discontinuities are prevalent in data, and often represent the key information of interest. Efficient representations for such discontinuities are important for many potential signal processing applications, including compression, but standard Fourier and wavelet representations fail to efficiently capture the structure of the discontinuities. These problems are most notable in image processing, where progress has been made in modeling and representing one-dimensional edge discontinuities along C^2 curves. Little work, however, has been done on efficient representations for higher dimensional functions, or for handling higher orders of smoothness in discontinuities. We consider a class of N-dimensional Horizon functions containing a C^K smooth singularity in $N-1$ dimensions, which serves as a boundary between two constant regions. We introduce the *surflet* representation for approximation and compression of functions in this class. Surflets allow a multiscale, piecewise polynomial approximation of the discontinuity. We propose a compression algorithm using surflets that comes within a logarithmic factor of the optimal rate-distortion performance for this function class. Equally important, this algorithm can be implemented using knowledge of the N-dimensional function, without explicitly estimating the $(N-1)$-dimensional discontinuity.

4.2 Problem definition and setup

We consider functions of N variables that contain a smooth discontinuity which is a function of $N-1$ variables. We denote vectors using boldface characters. Let $\mathbf{x} \in [0,1]^N$, and let x_i denote the i'th element of the vector $\mathbf{x}$. We denote the first $N-1$ elements of $\mathbf{x}$ by $\mathbf{y}$, i.e., $\mathbf{y} = [x_1, x_2, \cdots, x_{N-1}] \in [0,1]^{N-1}$.

4.2.1 Smoothness model for discontinuities

We first define the notion of smoothness that is used for modeling discontinuities. A function of $N-1$ variables is said to have smoothness of order $K > 0$, where $K = r + \alpha$, r is an integer, and $0 < \alpha \leq 1$ if the following criteria are met [37,38]:

- All iterated partial derivatives with respect to the $N-1$ directions up to order r exist and are continuous.

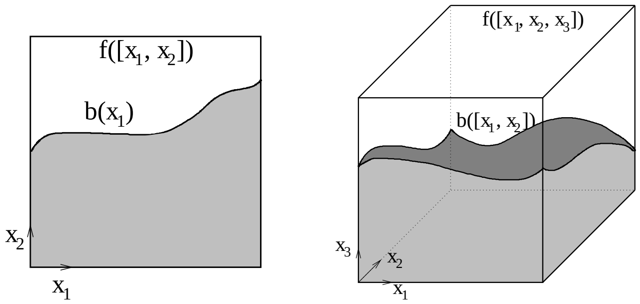


Figure 16: *Example Horizon-class functions for $N = 2$ and $N = 3$.*

- All such partial derivatives of order r satisfy a Lipschitz condition of order α (also known as a Hölder condition).[8]

We denote the space of such functions by $\mathcal{C}^K$. Observe that when K is an integer, $\mathcal{C}^K$ includes as a subset the traditional space C^K (where the function has $K = r + 1$ continuous partial derivatives).

4.2.2 Multidimensional Horizon-class functions

Let b be a function of $N - 1$ variables in $\mathcal{C}^K$ such that

$$b : [0, 1]^{N-1} \to [0, 1].$$

We define the function f of N variables such that

$$f : [0, 1]^N \to \{0, 1\}$$

according to the following:

$$f(\mathbf{x}) = \begin{cases} 1, & x_N \geq b(\mathbf{y}) \\ 0, & x_N < b(\mathbf{y}). \end{cases}$$

The function f is known as a Horizon-class function [39], where the smooth function b defines a horizon between values 0 and 1.

As shown in Figure 16, when $N = 2$ such a function can be interpreted as an image containing a smooth discontinuity that separates a black region (0-valued) below from a white region (1-valued) above. For $N = 3$, f represents a cube with a two-dimensional smooth surface cutting across the cube dividing it into two regions — black below the surface, and white above it.

4.2.3 Problem formulation

Our goal is to encode f, a Horizon-class function of N variables, and design a compression algorithm that accomplishes this with optimal asymptotic rate-distortion behavior. We use the squared-L_2 distortion metric to measure distortion between f and $\widehat{f}_R$, the approximation provided by the compression algorithm using R bits:

$$D_2(f, \widehat{f}_R) = \int_{\mathbf{x} \in [0,1]^N} (f - \widehat{f}_R)^2.$$

We emphasize that our algorithm approximates f in N dimensions. The approximation $\widehat{f}_R$, however, is Horizon-class — our algorithm implicitly provides an approximation $\widehat{b}_R$ to the smooth discontinuity b. That is,

$$\widehat{f}_R(\mathbf{x}) = \begin{cases} 1, & x_N \geq \widehat{b}_R(\mathbf{y}) \\ 0, & x_N < \widehat{b}_R(\mathbf{y}) \end{cases}$$

[8] A function $g \in$ Lip α if $|g(\mathbf{y} + \mathbf{h}) - g(\mathbf{y})| \leq C|\mathbf{h}|^\alpha$ for all $\mathbf{y}, \mathbf{h}$.

for some $\widehat{b}_R$. From the definition of N-dimensional Horizon-class functions, it is easy to see that

$$\begin{aligned} D_2(f, \widehat{f}_R) &= \int_{\mathbf{x}\in[0,1]^N} (f - \widehat{f}_R)^2 \\ &= \int_{\mathbf{y}\in[0,1]^{N-1}} |b - \widehat{b}_R| \\ &= D_1(b, \widehat{b}_R). \end{aligned} \tag{29}$$

Hence, optimizing for squared-L_2 distortion between f and $\widehat{f}_R$ is equivalent to optimizing for L_1 distortion between b and $\widehat{b}_R$.

4.2.4 Compression strategies and asymptotic performance

We assume that the coder is provided explicitly with the function f. As can be seen from the above formulation, all of the critical information about the function f is contained in the discontinuity b. One would expect any efficient coder to exploit such a fact. Methods through which this is achieved may vary.

One can imagine a coder that explicitly encodes b. Knowledge of b could be provided from an external "oracle" [30], or b could conceivably be estimated from the provided data f. Wavelets provide one efficient method for compressing the smooth function b. Cohen *et al.* [40] describe a tree-structured wavelet coder that can be used to compress b with rate-distortion performance

$$D_1(b, \widehat{b}_R) \sim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}.$$

The work of Clements [38] (extending Kolmogorov and Tihomirov [37]) establishes that this rate-distortion performance matches the metric entropy of the function class for b. We have also proved that no coder for f can outperform such a Horizon-class coder. Thus, this wavelet coder is optimal in terms of asymptotic rate-distortion performance on coding instances of f.

In practice, however, a coder is not provided with explicit information of b, and a method for estimating b from f may be difficult to implement. Estimates for b may also be quite sensitive to noise in the data. We propose a compression algorithm that operates directly on the N-dimensional data in f. This algorithm assembles an approximation $\widehat{f}_R$ that is Horizon-class (that is, it can be assembled using an estimate $\widehat{b}_R$), but it does not require explicit knowledge of b. We have proved that this algorithm achieves the following asymptotic rate-distortion performance

$$D_2(f, \widehat{f}_R) \sim \left(\frac{\log R}{R}\right)^{\frac{K}{N-1}},$$

and our current work strongly indicates that the $(\log R)$ factor can be removed to provide optimal rate-distortion performance. Our algorithm may also be more easily extended to similar function spaces containing smooth discontinuities. Our spatially localized approach, for example, allows for changes in the "transient variable" (assumed throughout to be x_N).

4.3 The surflet dictionary

In this section, we define a discrete dictionary of N-dimensional atoms, called *surflets*, which can be used to construct approximations to the Horizon-class function f. Each surflet consists of a dyadic hypercube containing a Horizon-class function, with a discontinuity defined by a smooth polynomial. Section 4.4 describes a specific compression algorithm for surflet approximations.

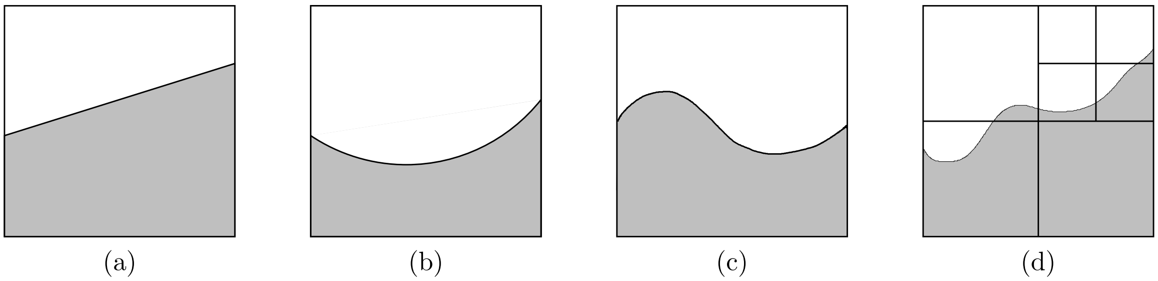


Figure 17: *Example surflets with $N = 2$, designed for smoothness (a) $K \in (1, 2]$, (b) $K \in (2, 3]$, (c) $K \in (3, 4]$. (d) Tiled surflet approximation.*

4.3.1 Motivation — Taylor's theorem

Construction of the surflet dictionary is motivated by the following property. If b is a function of $N-1$ variables in $\mathcal{C}^K$, Taylor's theorem states that

$$\begin{aligned} b(\mathbf{y}+\mathbf{h}) &= b(\mathbf{y}) + \frac{1}{1!}\sum_{i_1=1}^{N-1} b_{y_{i_1}}(\mathbf{y}) \cdot h_{i_1} \\ &+ \frac{1}{2!}\sum_{i_1,i_2=1}^{N-1} b_{y_{i_1}y_{i_2}}(\mathbf{y}) \cdot h_{i_1}h_{i_2} + \cdots \\ &+ \frac{1}{r!}\sum_{i_1,\cdots,i_r=1}^{N-1} b_{y_{i_1}\cdots y_{i_r}}(\mathbf{y}) \cdot h_{i_1}\cdots h_{i_r} \\ &+ O(\|\mathbf{h}\|^K), \end{aligned} \tag{30}$$

where $b_{y_1 \cdots y_\ell}$ refers to the iterated partial derivatives of b with respect to $y_1, \ldots, y_\ell$ in that order. Note that there are $(N-1)^\ell$ ℓ'th order derivative terms.

Thus, over a small domain, the function b is well approximated using an r'th order polynomial (where the polynomial coefficients correspond to the partial derivatives of b evaluated at $\mathbf{y}$). Clearly, then, one method of approximating b on a larger domain would be to assemble a *piecewise polynomial* approximation, where each polynomial is derived from the local Taylor approximation of b. Consequently, these piecewise polynomials may be used to assemble a Horizon-class approximation of the function f. Surflets provide the N-dimensional framework for constructing such approximations and can be implemented without explicit knowledge of b or its derivatives.

4.3.2 Definition

A *dyadic hypercube* $X \subseteq [0,1]^N$ at scale $j \in N$ is a domain that satisfies

$$X = [\beta_1 2^{-j}, (\beta_1+1)2^{-j}) \times \cdots \times [\beta_N 2^{-j}, (\beta_N+1)2^{-j}) \tag{31}$$

with $\beta_1, \beta_2, \ldots, \beta_N \in \{0, 1, \ldots, 2^j - 1\}$. We explicitly denote the $(N-1)$-dimensional hypercube *subdomain* of X as

$$Y = [\beta_1 2^{-j}, (\beta_1+1)2^{-j}) \times \cdots \times [\beta_{N-1} 2^{-j}, (\beta_{N-1}+1)2^{-j}). \tag{32}$$

The *surflet* $s(X;\cdot)$ is a Horizon-class function over the dyadic hypercube X. For $\mathbf{x} \in X$ with corresponding $\mathbf{y} = [x_1, x_2, \cdots, x_{N-1}]$, we have

$$s(X;\mathbf{x}) = \begin{cases} 1, & x_N \geq p(\mathbf{y}) \\ 0, & \text{otherwise}, \end{cases}$$

where the polynomial $p(\mathbf{y})$ is defined as follows

$$\begin{aligned} p(\mathbf{y}) &= s_0 + \sum_{i_1=1}^{N-1} s_{1,i_1} y_{i_1} + \sum_{i_1,i_2=1}^{N-1} s_{2,i_1,i_2} y_{i_1} y_{i_2} + \cdots \\ &+ \sum_{i_1,\cdots,i_r=1}^{N-1} s_{r,i_1,i_2,\ldots,i_r} y_{i_1} y_{i_2} \cdots y_{i_r}. \end{aligned}$$

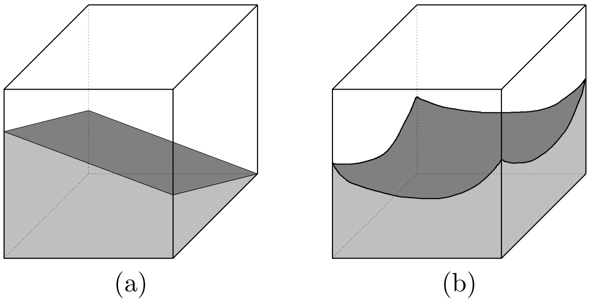


Figure 18: *Example surflets with $N = 3$, designed for smoothness (a) $K \in (1,2]$, (b) $K \in (2,3]$.*

The polynomial coefficients $\{s_{\ell,i_1,\ldots,i_\ell}\}_{\ell=0}^{r}$ are known as the *surflet coefficients*.[9] We note here that, in some cases, a surflet may be identically 0 or 1 over the entire domain X.

A surflet $s(X)$ approximates the function f over the dyadic hypercube X. One can cover the entire domain $[0,1]^N$ with a collection of dyadic hypercubes (possibly at different scales), and use surflets to approximate f over each of these smaller domains. For $N = 3$, these surflets together look like piecewise smooth "surfaces" approximating the function f. Therefore, we coin the term surflets to suggest "surface"-lets. Figure 17 illustrates a collection of surflets with $N = 2$ and shows an approximation obtained by combining localized surflets. Figure 18 illustrates a pair of surflets with $N = 3$.

4.3.3 Discretization

We obtain a discrete surflet dictionary by quantizing the set of allowable surflet coefficients. For $\ell \in \{0, 1, \ldots, r\}$, the surflet coefficient $s_{\ell,i_1,\ldots,i_\ell}$ at scale $j \in N$ is restricted to values $\{n \cdot \Delta_{\ell,j}\}_{n\in\mathbb{Z}}$, where the stepsize

$$\Delta_{\ell,j} = 2^{-(K-\ell)j}. \tag{33}$$

The necessary range for n may depend on the function b. However, all derivatives are locally bounded, and so the relevant discrete surflet dictionary is actually finite for any realization of f.

These quantization stepsizes are carefully chosen to ensure the proper fidelity of surflet approximations without requiring excess bitrate. The key idea is that *higher-order terms may be quantized with lesser precision*, without increasing the residual error term in the Taylor approximation (30). In fact, Kolmogorov and Tihomirov [37] implicitly used this concept to establish the metric entropy for the class $\mathcal{C}^K$.

4.4 Compression using surflets

4.4.1 Overview

Using surflets, we propose a tree-based multiresolution approach to approximate and encode f. The approximation is arranged on a 2^N-tree, where each node in the tree at scale j represents a hypercube of sidelength 2^{-j}. Every node is either a leaf node (hypercube), or has 2^N children nodes (children hypercubes that perfectly tile the volume of the parent hypercube). Each node in the tree is labeled with a surflet. Leaf nodes provide the actual approximation to the function f, while interior nodes are useful for predicting and encoding their descendants. This framework allows for *adaptive* approximation of f — many small surflets may be used at fine scales for more complicated regions, while few large surflets will suffice to encode simple regions of f (such as those containing all 0 or 1).

Section 4.4.2 discusses techniques for determining the proper surflet at each node. Section 4.4.3 mentions a method for pruning the tree depth according to the function f. Section 4.4.4 describes the performance of a simple surflet encoder acting only on the leaf nodes. Section 4.4.5 presents a more advanced surflet coder, using a top-down predictive technique to exploit the correlation among surflet coefficients.

4.4.2 Surflet Selection

Consider a node at scale j that corresponds to a dyadic hypercube X, and let Y be the $(N-1)$-dimensional subdomain of X as defined in (32).

[9] Because the ordering of terms $y_{i_1} y_{i_2} \cdots y_{i_\ell}$ within a monomial is not important, only $\binom{\ell+N-2}{\ell}$ monomial coefficients (not $(N-1)^\ell$) need to be encoded for order ℓ. We preserve the slightly redundant notation for ease of comparison with (30).

In a situation where the coder is provided with explicit information about the discontinuity b and its derivatives, determination of the surflet at this node may proceed as implied in Section 4.3. Specifically, the coder may construct the Taylor expansion of b about any point $\mathbf{y} \in Y$ and quantize the polynomial coefficients according to (33). To be precise, we choose

$$\mathbf{y} = [\beta_1 2^{-j}, \beta_2 2^{-j}, \ldots, \beta_{N-1} 2^{-j}]$$

and call this a *characteristic point.* We refer to the resulting surflet as the *quantized Taylor surflet.* Let p denote the Taylor polynomial centered at $\mathbf{y}$. From (30) we have

$$\max_{\mathbf{y}' \in Y} |b(\mathbf{y}') - p(\mathbf{y}')| = O(2^{-jK}).$$

If we let $\widehat{p}$ denote the polynomial with coefficients quantized according to (33), it can be proved that

$$\max_{\mathbf{y}' \in Y} |b(\mathbf{y}') - \widehat{p}(\mathbf{y}')| = O(2^{-jK}).$$

It follows that the squared-L_2 error of the quantized Taylor surflet approximation of f obeys

$$D_2(f, s(X)) = \int_{\mathbf{x} \in X} (f(\mathbf{x}) - s(X; \mathbf{x}))^2 = O(2^{-j(K+N-1)}). \tag{34}$$

As discussed in Section 4.2.4, a coder is not provided with explicit information of b. It is therefore important to mention a technique that can obtain a surflet estimate directly from the data f. We assume that there exists a technique to compute the squared-L_2 error between a given surflet $s(X)$ and the function f on the dyadic block,

$$D_2(f, s(X)) = \int_{\mathbf{x} \in X} (f(\mathbf{x}) - s(X; \mathbf{x}))^2.$$

In such a case, we may search the finite surflet dictionary for the minimizer of this error. We refer to the resulting surflet as the *L_2-best surflet.* This surflet will necessarily obey (34) as well. Sections 4.4.4 and 4.4.5 discuss the coding implications of using each type of surflet.

4.4.3 Organization of Surflet Trees

Given a method for assigning a surflet to each node, it is also necessary to determine the proper dyadic segmentation for the tree approximation. This may be accomplished using the CART (or Viterbi) algorithm in a process known as *tree-pruning* [39,41]. Tree-pruning acts in a bottom-up procedure, determining whether to prune the tree beneath each node (leaving it as a leaf node). Various criteria exist for making such a decision. In particular, the rate-distortion optimal segmentation may be obtained by minimizing the Lagrangian rate-distortion cost $D + \lambda R$ for a penalty term λ.

4.4.4 Leaf Encoding

An initial approach toward a surflet coder would encode a tree segmentation map denoting the location of leaf nodes, along with the quantized surflet coefficients at each leaf node. We have proved that, using either the quantized Taylor surflets or the L_2-best surflets, the asymptotic rate-distortion performance of this coder obeys

$$D_2(f, \widehat{f}_R) \sim \left(\frac{\log R}{R}\right)^{\frac{K}{N-1}}.$$

Thus this simple coder is *near-optimal* in terms of rate-distortion performance. In addition, it can be implemented using L_2-best surflets, which are derived directly from the function f.

4.4.5 Top-down Predictive Encoding

Achieving the optimal rate-distortion performance requires a slightly more sophisticated coder. In particular, the above encoding technique fails to exploit the correlation among nearby surflets. In this section, we briefly describe a top-down surflet coder that achieves the optimal performance by predicting surflet parameters from previously encoded values.

Suppose we have a tree segmentation, where each node is labeled with a quantized Taylor surflet.[10] We encode the entire tree starting with the root node, and proceeding from the top down. Given a quantized surflet $s(X_j)$ at an interior node at scale j, we may encode its children surflets (scale $j+1$) according to the following procedure.

[10] We later address the related procedure using L_2-best surflets.

1) (Parent-child prediction). Let Y_j be the subdomain of X_j, and let $Y_{j+1} \subset Y_j$ be the single dyadic hypercube at scale $j+1$ that shares the same characteristic point with Y_j. Thus, for each surflet $s(X_{j+1})$ with subdomain on Y_{j+1}, every coefficient of $s(X_{j+1})$ is also a surflet coefficient of (the previously encoded) $s(X_j)$, but more precision must be provided to achieve (33). The coder should provide the necessary bits.

2) (Child-neighbor prediction). We now use surflets encoded on Y_{j+1} (in Step 1) to predict the surflet coefficients for each of the remaining hypercube children of X_j. We omit the precise details but note that this prediction operates according to (30), with $||\mathbf{h}|| \sim 2^{-(j+1)}$.

We have proved that the number of bits required to encode each surflet using the above procedure is constant with respect to the scale j. It then follows that the asymptotic rate-distortion performance of this coder obeys

$$D_2(f, \widehat{f}_R) \sim \left(\frac{1}{R}\right)^{\frac{K}{N-1}}.$$

This coder achieves the *optimal* asymptotic rate-distortion performance. The additional information encoded at interior nodes provides the key to efficiently encoding the leaf nodes.

The encoding procedure as described above assumes a quantized Taylor surflet for each node. Our current work strongly indicates that a similar predictive coding scheme can be developed using L_2-best surflets, and that this scheme will achieve the same optimal performance. The development of such a technique, and proof of its performance, will be discussed at the conference.

4.5 Conclusions and Future Work

The surflet-based compression framework that we describe provides a sparse representation of higher dimensional functions with smooth discontinuities. We present a tractable method to encode piecewise smooth polynomials to approximate such functions. Some of the insights that we gain, namely, quantization of higher-order terms with lesser precision, and predictive coding to decrease bitrate, can be used to solve to more sophisticated compression problems. In addition, our method only requires knowledge of the higher dimensional function, not the smooth discontinuity.

5 Publications supported by this grant

R. G. Baraniuk, M. A. Davenport, R. A. DeVore, and M. B. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, pp. 253–263, Dec. 2008.

D. Baron and R. G. Baraniuk, "Faster sequential universal coding via block partitioning," *IEEE Transactions on Information Theory*, vol. 52, pp. 1708–1710, Apr. 2006.

W. L. Chan, H. Choi, and R. G. Baraniuk, "Coherent multiscale image processing using quaternion wavelets," *IEEE Transactions on Image Processing*, vol. 17, p. 1069–1082, Jul. 2008.

W. L. Chan, H. Choi, and R. G. Baraniuk, "Directional hypercomplex wavelets for multidimensional signal analysis and processing," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, (Montreal), pp. 996–999, May 2004.

W. L. Chan, H. Choi, and R. G. Baraniuk, "Quaternion wavelets for image analysis and processing," in *IEEE International Conference on Image Processing*, (Singapore), pp. 3057–3060, Oct. 2004.

V. Chandrasekaran, M. B. Wakin, D. Baron, and R. G. Baraniuk, "Compression of higher dimensional functions containing smooth discontinuities," in *Conference on Information Sciences and Systems*, (Princeton), Mar. 2004.

V. Chandrasekaran, M. B. Wakin, D. Baron, and R. G. Baraniuk, "Representation and compression of multidimensional piecewise functions using surflets," *IEEE Transactions on Information Theory*, vol. 55, p. 374–400, Jan. 2009.

V. Chandrasekaran, M. B. Wakin, D. Baron, and R. G. Baraniuk, "Surflets: A sparse representation for multidimensional functions containing smooth discontinuities," in *IEEE Symposium on Information Theory*, (Chicago), Jul. 2004.

H. Choi and R. G. Baraniuk, "Multiple wavelet basis image denoising using besov ball projections," *IEEE Signal Processing Letters*, vol. 11, pp. 717–720, Sept. 2004.

H. Choi and R. G. Baraniuk, "Multiscale manifold representation and modeling," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, (Philadelphia), Mar. 2005.

M. A. Davenport, R. G. Baraniuk, and C. D. Scott, "Learning minimum volume sets with support vector machines," in *IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, (Maynooth, Ireland), Sep. 2006.

S. Lavu, H. Choi, and R. G. Baraniuk, "Estimation-quantization geometry coding using normal meshes," in *Data Compression Conference*, (Snowbird, Utah), Mar. 2003.

R. Neelamani, H. Choi, and R. G. Baraniuk, "Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Transactions on Signal Processing*, vol. 52, pp. 418–433, Feb, 2004.

J. K. Romberg, M. B. Wakin, H. Choi, and R. G. Baraniuk, "A geometric hidden markov tree wavelet model," in *International Symposium on Optical Science and Technology*, (San Diego), Aug. 2003.

I. W. Selesnick, R. G. Baraniuk, and N. G. Kingsbury, "The dual-tree complex wavelet transform," *IEEE Signal Processing Magazine*, vol. 22, pp. 123–151, Nov. 2005.

M. B. Wakin, D. Donoho, H. Choi, and R. G. Baraniuk, "The multiscale structure of non-differentiable image manifolds," in *Wavelets XI in SPIE International Symposium on Optical Science and Technology*, (San Diego), SPIE, SPIE, Aug. 2005.

M. B. Wakin, J. K. Romberg, H. Choi, and R. G. Baraniuk, "Geometric methods for wavelet-based image compression," in *International Symposium on Optical Science and Technology*, (San Diego), Aug. 2003.

M. B. Wakin, J. K. Romberg, H. Choi, and R. G. Baraniuk, "Wavelet-domain approximation and compression of piecewise smooth images," *IEEE Transactions on Image Processing*, vol. 15, pp. 1071–1087, May 2006.

6 Professional personnel

Principle Investigators
Richard G. Baraniuk, Hyeokho Choi

Postdoctoral Research Associates
Dror Baron, Rutger van Spaendonck

Graduate Student Research Assistants
Wai Lam Chan, Ryan King, Jason Laska, William Mantzel, Justin Romberg, Christopher Rozell, Shriram Sarvotham, Michael Wakin

References

[1] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding.* Englewood Cliffs, NJ: Prentice Hall, 1995.

[2] I. Selesnick, R. Baraniuk, and N. Kingsbury, "The Dual-tree Complex Wavelet Transform," *IEEE Sig. Proc. Mag.*, pp. 123–151, November 2005.

[3] N. G. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *J. App. Harm. Analysis*, vol. 10, pp. 234–253, May 2001.

[4] I. W. Selesnick and K. Y. Li, "Video denoising using 2-D and 3-D dual-tree complex wavelet transforms," in *Proc. of SPIE Wavelets X*, vol. 76, (San Diego, CA), August 4-8 2003.

[5] T. Bülow and G. Sommer, "A novel approach to the 2-D analytic signal," in *CAIP*, (Ljubljana, Slovenia), 1999.

[6] S. L. Hahn, "Multidimensional complex signals with single-orthant spectra," in *Proceedings of the IEEE*, vol. 80, pp. 1287–1300, August 1992.

[7] T. Bülow, *Hypercomplex Spectral Signal Representations for the Processing and Analysis of Images.* PhD thesis, Christian Albrechts University, Kiel, Germany, 1999.

[8] I. L. Kantor and A. S. Solodovnikov, *Hypercomplex Numbers.* Springer-Verlag, 1989.

[9] J. Romberg, H. Choi, and R. Baraniuk, "Multiscale edge grammars for complex wavelet transforms," in *Proc. of Intl. Conf. on Image Processing*, (Thessaloniki, Greece), pp. 614–617, Oct. 2001.

[10] J. Romberg, M. Wakin, H. Choi, and R. Baraniuk, "A geometric hidden Markov tree wavelet model," in *Proc. of SPIE Wavelets X*, (San Diego, CA), August 2003.

[11] M. Wakin, J. Romberg, H. Choi, and R. Baraniuk, "Rate-Distortion Optimized Image Compression using Wedgelets," in *IEEE International Conference on Image Processing*, September 2002.

[12] W. L. Chan, H. Choi, and R. G. Baraniuk, "Coherent multiscale image processing using quaternion wavelets," *IEEE Transactions on Image Processing*, vol. 17, p. 1069–1082, Jul. 2008.

[13] W. L. Chan, H. Choi, and R. G. Baraniuk, "Quaternion wavelets for image analysis and processing," in *Proc. of IEEE Int. Conf. on Image Processing*, vol. 5, (Singapore), pp. 3057–3060, Oct. 2004.

[14] http://sampl.ece.ohio-state.edu/data/motion/Heart/.

[15] V. Argyriou and T. Vlachos, "Using gradient correlation for sub-pixel motion estimation of video sequences," *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 329–332, May 2004.

[16] M. Hemmendroff, M. Anderson, T. Kronander, and H. Knutsson, "Phase-based multidimensional volume registration," *IEEE Trans. Medical Imaging*, vol. 21, pp. 1536–1543, December 2002.

[17] N. Kingsbury, "Dual Tree Complex Wavelets (HASSIP Workshop)." http://www.eng.cam.ac.uk/˜ngk, September 2004.

[18] M. Felsberg, "Optical flow estimation from monogenic phase.," in *1st International Workshop on Complex Motion (IWCM)*, vol. LNCS 3417, (Günzburg, Tyskland), October 2004. In press.

[19] E. Bayro-Corrochano, "The Theory and Use of the Quaternion Wavelet Transform," *J. Mathematical Imaging and Vision*, vol. 24, no. 1, pp. 19–35, 2006.

[20] J. Zhou, Y. Xu, and X. K. Yang, "Quaternion wavelet phase based stereo matching for uncalibrated images," *Pattern Recognition letters*, vol. 28, pp. 1509–1522, March 2007.

[21] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[22] W. L. Chan, H. Choi, and R. Baraniuk, "Directional hypercomplex wavelets for multidimensional signal analysis and processing," *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 996–999, May 2004.

[23] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

[24] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Space-frequency quantization for wavelet image coding," *IEEE Trans. Image Processing*, vol. 6, no. 5, pp. 677–693, 1997.

[25] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using hidden Markov models," *IEEE Trans. Signal Processing*, vol. 46, pp. 886–902, Apr. 1998.

[26] H. Choi, J. K. Romberg, R. G. Baraniuk, and N. G. Kingsbury, "Hidden Markov tree modeling of complex wavelet transforms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. — ICASSP '00*, (Istanbul, Turkey), June 2000.

[27] H. Choi and R. G. Baraniuk, "Multiscale image segmentation using wavelet-domain hidden Markov models," *IEEE Trans. Image Processing*, vol. 10, pp. 1309–1321, Sept. 2001.

[28] E. Candés and D. Donoho, "Curvelets – a surpringly effective nonadaptive representation for objects with edges," in *Curve and Surface Fitting* (A. C. et. al, ed.), Vanderbilt University Press, 1999.

[29] E. L. Pennec and S. Mallat, "Image compression with geometrical wavelets," in *IEEE Int. Conf. on Image Proc. – ICIP '01*, (Thessaloniki, Greece), Oct. 2001.

[30] M. N. Do, P. L. Dragotti, R. Shukla, and M. Vetterli, "On the compression of two-dimensional piecewise smooth functions," in *IEEE Int. Conf. on Image Proc. – ICIP '01*, (Thessaloniki, Greece), Oct. 2001.

[31] R. Shukla, P. L. Dragotti, M. Do, and M. Vetterli, "Rate distortion optimized tree structured compression algorithms," *IEEE Trans. Image Processing*, submitted for publication.

[32] M. B. Wakin, J. K. Romberg, H. Choi, and R. G. Baraniuk, "Rate-distortion optimized image compression using wedgelets," in *IEEE Int. Conf. on Image Proc. – ICIP '02*, (Rochester, NY), September 2002.

[33] N. G. Kingsbury, "Image processing with complex wavelets," *Phil. Trans. Royal Society London A*, vol. 357, pp. 2543–2560, Sept. 1999.

[34] J. K. Romberg, M. B. Wakin, and R. G. Baraniuk, "Multiscale wedgelet image analysis: fast decompositions and modeling," in *IEEE Int. Conf. on Image Proc. – ICIP '02*, (Rochester, NY), Sept. 2002.

[35] I. W. Selesnick, "Hilbert transform pairs of wavelet bases," *IEEE Signal Processing Lett.*, vol. 8, pp. 170–173, June 2001.

[36] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–285, Feb. 1989.

[37] A. N. Kolmogorov and V. M. Tihomirov, "ϵ-entropy and ϵ-capacity of sets in functional spaces," *Amer. Math. Soc. Transl. (Ser. 2)*, vol. 17, pp. 277–364, 1961.

[38] G. F. Clements, "Entropies of several sets of real valued functions," *Pacific J. Math.*, vol. 13, pp. 1085–1095, 1963.

[39] D. L. Donoho, "Wedgelets: Nearly-minimax estimation of edges," *Annals of Stat.*, vol. 27, pp. 859–897, 1999.

[40] A. Cohen, W. Dahmen, I. Daubechies, and R. DeVore, "Tree approximation and optimal encoding," *J. App. Comp. Harmonic Analysis*, vol. 11, pp. 192–226, 2001.

[41] J. K. Romberg, M. B. Wakin, and R. G. Baraniuk, "Multiscale Geometric Image Processing," in *SPIE Vis. Comm. and Image Proc.*, (Lugano, Switzerland), July 2003.